

**Řešení problémů vlastních čísel s  
aplikacemi v molekulové dynamice**  
**Solution of the eigenvalue problem  
with application to molecular  
dynamics**

# Zadání bakalářské práce

Student:

**Martin Mrovec**

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

1103R031 Výpočetní matematika

Téma:

Řešení problému vlastních čísel s aplikacemi v molekulové dynamice  
Solution of the eigenvalue problem with application to molecular  
dynamics

Zásady pro vypracování:

Cílem diplomové práce je pochopit motivaci problému vlastních čísel v oblastech výzkumu souvisejících s aktivitou na Katedře aplikované matematiky VŠB (zejména molekulární simulace), prostudovat základní techniky výpočtu vlastních čísel hermitovsky symetrických pozitivně definitních matic, naprogramovat je a porovnat je na jednoduchých testovacích příkladech.

Postup práce

1. Prostudovat doporučenou literaturu.
2. Vyhledat aplikace.
3. Naprogramovat vybrané metody v MATLABU.
4. Provést numerické experimenty.
5. Zpracovat výsledky.

The student should follow the following scheme:

1. Read the references.
2. Find applications related to the research of the department.
3. Code chosen algorithms in MATLAB.
4. Carry out numerical experiments.
5. Process the results

Seznam doporučené odborné literatury:

1. Y. Saad, Numerical Methods for Large eigenvalue Solvers, SIAM, Philadelphia 2010.
2. Internetové zdroje

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **prof. RNDr. Zdeněk Dostál, DSc.**

Datum zadání: 16.11.2012

Datum odevzdání: 07.05.2013



---

doc. RNDr. Jiří Bouchala, Ph.D.  
*vedoucí katedry*

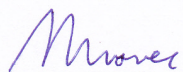


---

prof. RNDr. Václav Snášel, CSc.  
*děkan fakulty*

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7. května 2013

.....

Na tomto místě bych chtěl poděkovat prof. RNDr. Zdeňku Dostálovi, DSc. za vedení této bakalářské práce. Poděkování patří i doc. RNDr. Renému Kalusovi, Ph.D. za pomoc, kterou mi poskytl při zpracování tématu molekulové dynamiky. Děkuji také své nejbližší rodině za prostředí, které mi vytvořila k napsání této práce.

## **Abstrakt**

Práce se zabývá problematikou vlastních čísel a vlastních vektorů převážně hermitovských a reálných symetrických matic. Obsahuje úvod do teorie vlastních čísel a vlastních vektorů. Hlavní částí textu je popis několika metod jejich hledání včetně následné implementace v Matlabu. Následně jsou tyto metody porovnány a je diskutována vhodnost jejich využití. Na závěr jsou probrány dvě oblasti, kde se problémy vlastních čísel vyskytují. Jedná se konkrétně o mechanické kmitání a molekulovou dynamiku. Ve většině kapitol jsou uvedeny názorné příklady sloužící k lepšímu pochopení dané problematiky.

**Klíčová slova:** matice, vlastní číslo, vlastní vektor

## **Abstract**

The thesis deals with eigenvalue and eigenvector problems mainly of the Hermitian matrices and real symmetric matrices. It contains an introduction to the theory of eigenvalues and eigenvectors. The main part of the text describes several methods of their search and includes an implementation of them in Matlab. After that, these methods are compared, and suitability of their usage is discussed. Finally, two areas, where eigenvalue problems occur, are explained. These are mechanical vibrations and molecular dynamics. In most chapters there are given examples which serve for better understanding of the issue.

**Keywords:** matrix, eigenvalue, eigenvector

## Seznam použitých zkratk a symbolů

$A^{-1}$	– inverzní matice
$A^H$	– konjugovaná transpozice matice
$A^T$	– transpozice matice
$\mathbb{C}$	– množina komplexních čísel
$E$	– celková energie
$F$	– síla v N
$I$	– jednotková matice
$O$	– nulová matice
$\mathbb{R}$	– množina reálných čísel
$V(x)$	– potenciální energie v bodě $x$
$\bar{b}$	– komplexně sdružené číslo k číslu $b$
$\det(A)$	– determinant matice $A$
$f$	– frekvence v $s^{-1}$
$\hbar$	– redukovaná Planckova konstanta
$k$	– tuhost pružiny v $N \cdot m^{-1}$
$m$	– hmotnost v kg
$p(\lambda)$	– charakteristický polynom s neznámou $\lambda$
$r$	– vektor polohy částice
$u(x)$	– interakční potenciál v bodě $x$
$\dot{y}$	– první derivace proměnné $y$ podle času
$\ddot{y}$	– druhá derivace proměnné $y$ podle času
$\Gamma(S)$	– součet druhých mocnin mimodiagonálních prvků symetrické matice $S$
$\lambda$	– vlastní číslo matice
$\rho(A)$	– spektrální poloměr matice $A$
$\sigma(A)$	– spektrum matice $A$
$\psi(x)$	– prostorová vlnová funkce v bodě $x$
$\omega$	– úhlová frekvence v $s^{-1}$

## Obsah

<b>1</b>	<b>Úvod</b>	<b>5</b>
<b>2</b>	<b>Vlastní čísla a vlastní vektory</b>	<b>6</b>
<b>3</b>	<b>Metody hledání vlastních čísel</b>	<b>8</b>
3.1	Charakteristický polynom . . . . .	8
3.2	Mocninná metoda . . . . .	13
3.3	Jacobiho metoda . . . . .	19
3.4	QR metoda . . . . .	23
3.5	Srovnání metod . . . . .	27
<b>4</b>	<b>Aplikace metod hledání vlastních čísel</b>	<b>29</b>
4.1	Mechanické kmity . . . . .	29
4.2	Molekulová dynamika . . . . .	33
<b>5</b>	<b>Závěr</b>	<b>37</b>
<b>6</b>	<b>Reference</b>	<b>38</b>
	<b>Přílohy</b>	<b>38</b>
<b>A</b>	<b>Zdrojové kódy</b>	<b>39</b>
A.1	Jacobiho metoda . . . . .	39
A.2	QR metoda . . . . .	41
A.3	Mocninná metoda . . . . .	42
A.4	Mocninná metoda s posunem spektra . . . . .	43
A.5	Inverzní mocninná metoda . . . . .	44



## Seznam tabulek

1	Závislost počtu iterací mocninné metody na podílu v abs. hodnotě dvou největších vlastních čísel . . . . .	27
2	Závislost počtu iterací na použité metodě . . . . .	28

## Seznam obrázků

1	Závaží na pružině . . . . .	30
2	Soustava závaží a pružin . . . . .	31
3	Graf Lennard-Jonesova potenciálu . . . . .	35

## Seznam výpisů zdrojového kódu

1	Implementace Jacobiho metody v Matlabu . . . . .	39
2	Implementace QR metody v Matlabu . . . . .	41
3	Implementace mocninné metody v Matlabu . . . . .	42
4	Implementace mocninné metody s posunem spektra v Matlabu . . . . .	43
5	Implementace nalezení vlastního čísla nejbližšího zadané hodnotě . . . . .	44

## 1 Úvod

Úlohy spojené s hledáním vlastních čísel jsou v současné době nedílnou součástí výzkumu spojeného s aplikovanou matematikou. Může se na první pohled zdát, že se tento výzkum týká pouze oblasti techniky. Ve skutečnosti nalézají metody hledání vlastních čísel uplatnění např. i v ekonomii. S hledáním vlastních čísel a vektorů se setkáváme každodenně při používání internetových vyhledávačů. Z některých úloh vycházejí matice o dimenzích v řádech miliard i více. Pro určování spektra takových matic je zapotřebí neustále vyvíjet nové metody, které zvyšují rychlost výpočtu, ale zachovávají požadovanou přesnost. Jedním z pracovišť, kde probíhá výzkum molekulových simulací, jenž souvisí s vlastními čísly, je katedra aplikované matematiky působící na Vysoké škole báňské. Tato práce přináší pohled do problematiky vlastních čísel a je rozdělena do tří částí.

První část je úvodem do spektrální teorie. Obsahuje definici vlastních čísel a vlastních vektorů. Dále jsou uvedeny jejich obecné vlastnosti a také speciální vlastnosti v případě hermitovských matic.

Druhá část se zabývá jednotlivými metodami výpočtu vlastních čísel. Zahrnuje teorii spojenou s danými metodami, popis algoritmu a implementaci v Matlabu. Jejich funkčnost je také demonstrována na jednoduchých příkladech. Na závěr druhé části jsou jednotlivé metody porovnány a je diskutována možnost jejich využití.

Třetí část je seznámením s několika oblastmi, kde se vyskytuje problém hledání vlastních čísel.

## 2 Vlastní čísla a vlastní vektory

Mějme čtvercovou matici  $A \in \mathbb{C}^{n \times n}$ . Pak vlastním vektorem  $v \in \mathbb{C}^n$  rozumíme takový nenulový vektor, pro který platí:

$$Av = \lambda v, \quad (2.1)$$

kde  $\lambda \in \mathbb{C}$  nazveme vlastním číslem matice  $A$ . Jinak řečeno, pokud  $A$  je maticí lineárního zobrazení  $\mathfrak{V} : \mathbb{C}^n \rightarrow \mathbb{C}^n$ , pak vlastní vektor je takový nenulový vektor, jehož směr se transformací nemění. Mění se pouze jeho velikost. Poměr mezi velikostí transformovaného a původního vektoru je roven vlastnímu číslu. Rovnici (2.1) můžeme upravit do tvaru

$$(A - \lambda I)v = o, \quad (2.2)$$

kde  $I$  je jednotková matice o rozměrech  $n \times n$  a  $o$  je nulový vektor dimenze  $n$ . Jelikož  $v$  je nenulový vektor, nemůže být zobrazení  $(A - \lambda I)$  prosté, a tedy jeho matice je singulární. O singulárních maticích víme, že jejich determinant je roven nule (viz. [5]):

$$\det(A - \lambda I) = 0. \quad (2.3)$$

Výše uvedený determinant nazýváme charakteristickým polynomem s neznámou  $\lambda$

$$p(\lambda) = a_n \lambda^n + a_{n-1} \lambda^{n-1} + \dots + a_1 \lambda + a_0, \quad (2.4)$$

kořeny tohoto polynomu jsou právě vlastní čísla matice  $A$ . Některé kořeny charakteristického polynomu mohou být násobné, potom hovoříme o algebraické násobnosti vlastního čísla. Charakteristický polynom neobsahuje pouze informaci o vlastních číslech, např. koeficient  $a_0$  je přímo roven determinantu matice  $A$ . Tuto skutečnost není nutné složitě dokazovat, stačí si jen uvědomit, jak probíhá výpočet  $\det(A - \lambda I)$ .

Množina všech vlastních čísel matice  $A$  se nazývá spektrum matice  $A$  a značíme ji  $\sigma(A)$ . Dominantním vlastním číslem rozumíme v absolutní hodnotě největší vlastní číslo. Jeho absolutní hodnotu nazýváme též spektrálním poloměrem matice  $A$  a značíme  $\rho(A)$ . Dále pro vlastní čísla platí, že jejich součet je roven součtu diagonálních prvků (stopě) matice  $A$  a jejich součin je roven determinantu  $A$  (důkaz lze nalézt např. v [5]), tedy

$$\sum_{i=1}^n \lambda_i = \sum_{i=1}^n A_{ii} \quad (2.5)$$

$$\prod_{i=1}^n \lambda_i = \det A. \quad (2.6)$$

Pokud je algebraická násobnost vlastního čísla větší než 1, musí být v součtu (resp. v součinu) započteno tolikrát, kolik je právě jeho násobnost.

Označme  $A^H \in \mathbb{C}^{m \times n}$  takovou matici, pro kterou platí

$$A_{ij}^H = \overline{A_{ji}} \quad \forall i, j \in \{1, 2, \dots, n\}, \quad (2.7)$$

kde označení  $\overline{A}_{ji}$  znamená komplexně sdružené číslo k číslu  $A_{ji}$ . Tato matice se nazývá konjugovanou transpozicí matice  $A \in \mathbb{C}^{n \times m}$ . Čtvercová matice, jež splňuje rovnost

$$A^H = A, \quad (2.8)$$

se nazývá maticí hermitovskou. Speciálním případem hermitovských matic jsou matice symetrické, a to v případě, kdy všechny prvky matice  $A$  jsou reálné. Pro hermitovské matice platí následující věta:

**Věta 2.1** *Nechť  $A$  je hermitovská matice. Potom platí*

$$\forall \lambda_i, i \in \{1, 2, \dots, n\} : \lambda_i \in \mathbb{R}.$$

**Důkaz.** Z definice hermitovské matice přímo vyplývá, že  $A = A^H$ . Dále všechna vlastní čísla a jim odpovídající vlastní vektory splňují rovnost (2.1). Označme  $a = v^H v$ , kde  $v$  je vlastní vektor matice  $A$ . Není těžké si uvědomit, že  $a$  je reálné číslo. Platí totiž

$$a = \sum_{t=1}^n \overline{v_t} v_t = \sum_{t=1}^n (\overline{k_t + l_t i})(k_t + l_t i) = \sum_{t=1}^n (k_t - l_t i)(k_t + l_t i) = \sum_{t=1}^n (k_t^2 + l_t^2), \quad k, l \in \mathbb{R}.$$

Dále označme  $b = v^H A v$ . Určíme, čemu je rovno  $\bar{b}$ :

$$\bar{b} = (v^H A v)^H = v^H A^H (v^H)^H = v^H A (v^H)^H = v^H A v = b.$$

Z této rovnosti plyne, že  $b \in \mathbb{R}$ . Číslo  $b$  můžeme zapsat i jinak:

$$b = v^H A v = v^H \lambda v = \lambda v^H v.$$

Porovnáme-li nyní  $a$  a  $b$ , zjistíme, že platí rovnost

$$b = \lambda a.$$

A jelikož čísla  $a$  a  $b$  jsou reálná, musí být  $\lambda$  také reálné číslo. ■

### 3 Metody hledání vlastních čísel

Existuje mnoho způsobů, jak nalézt vlastní čísla matice  $A$ . Jejich hodnoty lze určit jak analyticky, např. s využitím charakteristického polynomu, tak numericky, kde se používají různé iterační metody. Již u matic s rozměry většími než  $4 \times 4$  se však s analytickým řešením dostáváme do problému, protože nejsme schopni obecně najít kořeny polynomu stupně 5 nebo vyššího. V technické praxi se zpravidla setkáváme s rozměrnějšími maticemi, ve většině případů se tedy přikláníme k metodám numerickým. V této kapitole si ukážeme několik metod (převážně numerických) určených k výpočtu vlastních čísel a vektorů.

#### 3.1 Charakteristický polynom

Jak již bylo uvedeno v kapitole 2, kořeny charakteristického polynomu matice  $A$  jsou její vlastní čísla. Charakteristický polynom lze jednoduše vypočítat přímo z definice, tedy

$$p(\lambda) = \det(A - \lambda I). \quad (3.1)$$

Pro menší matice je tato metoda vhodná, ale s rostoucími rozměry matice velmi rychle narůstá náročnost výpočtu determinantu. Využijeme-li např. rozvoje podle sloupce, musíme u matice o rozměrech  $n \times n$  vypočítat  $\frac{n!}{2}$  krát determinant matice  $2 \times 2$ . Bohužel zde nepřichází v úvahu ani využití Gaussovy eliminace, jelikož na diagonále se nachází neznámá  $\lambda$  a prvky pod diagonálou by bylo obtížné vynulovat (bylo by možné např. přeuspořádat řádky, ale neznámá  $\lambda$  se bude v matici neustále vyskytovat). Je potřeba tedy nalézt alternativní způsob nalezení charakteristického polynomu.

##### 3.1.1 Krylovova metoda

Krylovova metoda vychází z Cayleyho-Hamiltonovy věty (viz. [4],[8]):

**Věta 3.1** *Každá čtvercová matice  $A$  nad komutativním tělesem je kořenem svého charakteristického polynomu, tedy*

$$p_A(A) = O. \quad (3.2)$$

**Důkaz.** Nejprve je nutné poznamenat, že množiny  $\mathbb{R}$  a  $\mathbb{C}$  komutativními tělesy jsou. Jak operace  $+$ , tak operace  $\times$  jsou totiž v obou množinách komutativní.

Při důkazu vyjdeme ze skutečnosti, že každou čtvercovou matici  $A$  lze rozložit následujícím způsobem:

$$A = SJS^{-1}, \quad (3.3)$$

kde  $S$  je regulární matice a  $J$  je matice Jordanova (viz. [5], [1]). Jordanovu matici lze zapsat blokově

$$J = \begin{pmatrix} J_{\lambda_1} & & & \\ & J_{\lambda_2} & & \\ & & \ddots & \\ & & & J_{\lambda_p} \end{pmatrix}, \quad J_{\lambda_i} = \begin{pmatrix} \lambda_i & 1 & 0 & \dots & 0 \\ 0 & \lambda_i & 1 & & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & & \ddots & \lambda_i & 1 \\ 0 & \dots & \dots & 0 & \lambda_i \end{pmatrix}. \quad (3.4)$$

$\lambda_i$ ,  $i \in \{1, 2, \dots, p\}$  jsou všechna vlastní čísla matice  $A$  (některá mohou být shodná). Každému vlastnímu číslu přísluší Jordanův blok  $J_{\lambda_i}$ . Ten má dimenzi rovnou maximálnímu počtu lineárně nezávislých vlastních vektorů, které tomuto vlastnímu číslu přísluší. Podívejme se, jak vypadají mocniny matice  $A$ :

$$A^2 = SJS^{-1}SJS^{-1} = SJ^2S^{-1} \Rightarrow A^n = SJ^nS^{-1}, \quad n \in \mathbb{N}. \quad (3.5)$$

Pokud tedy dosadíme matici  $A$  do jejího charakteristického polynomu, dostaneme

$$p_A(A) = a_n A^n + a_{n-1} A^{n-1} + \dots + a_1 A + a_0 I \quad (3.6)$$

Úpravou s využitím (3.5) získáme

$$p_A(A) = a_n SJ^n S^{-1} + a_{n-1} SJ^{n-1} S^{-1} + \dots + a_1 SJS^{-1} + a_0 SIS^{-1} \quad (3.7)$$

$$p_A(A) = S(a_n J^n + a_{n-1} J^{n-1} + \dots + a_1 J + a_0 I)S^{-1}. \quad (3.8)$$

Stačí dokázat, že Jordanova matice  $J$  příslušná matici  $A$  je kořenem charakteristického polynomu matice  $A$ . Není těžké si uvědomit, že jednotlivé Jordanovy bloky se navzájem neovlivňují. Umocníme-li totiž Jordanovu matici, vypadá výsledek takto:

$$J^n = \begin{pmatrix} J_{\lambda_1}^n & & & \\ & J_{\lambda_2}^n & & \\ & & \ddots & \\ & & & J_{\lambda_p}^n \end{pmatrix}. \quad (3.9)$$

Úloha se nám tedy redukuje na důkaz toho, že každý Jordanův blok Jordanovy matice  $J$  příslušné matici  $A$  je kořenem charakteristického polynomu matice  $A$ . Vyšetříme nejprve, jestli je Jordanův blok kořenem svého charakteristického polynomu. Jordanův blok  $J_{\lambda_i}$  má všechna vlastní čísla stejná. Jsou jimi právě  $\lambda_i$ . Charakteristický polynom Jordanova bloku má tvar

$$p_{J_{\lambda_i}}(\lambda) = (\lambda - \lambda_i)^n, \quad (3.10)$$

Dosadíme-li do něj samotný Jordanův blok, dostaneme

$$p_{J_{\lambda_i}}(J_{\lambda_i}) = (J_{\lambda_i} - \lambda_i I)^n, \quad (3.11)$$

neboli

$$p_{J_{\lambda_i}}(J_{\lambda_i}) = (N)^n, \quad (3.12)$$



kde matice  $N$  má na superdiagonále jedničky, zbylé prvky jsou nuly. Pokud umocníme takovou matici na  $i$ , posunou se jedničky o  $i - 1$  pozic doprava. Jednička, která leží v posledním sloupci bude nahrazena nulou. My zde matici  $N$  umocňujeme na  $n$ , takže všechny jedničky se posunou o  $n - 1$  pozic doprava, z čehož vyplývá, že zůstane pouze nulová matice  $O$ . To znamená, že Jordanův blok je kořenem svého charakteristického polynomu. Tím lze však beze zbytku vydělit charakteristický polynom matice  $A$ , takže jej lze před něj vytknout. Proto je i kořenem charakteristického polynomu matice  $A$ , což jsme chtěli dokázat. ■

Rovnost (3.2) můžeme zapsat také následujícím způsobem:

$$(-1)^n A^n + a_{n-1} A^{n-1} + \dots + a_1 A + a_0 I = O. \quad (3.13)$$

Vydělme rovnici (3.13)  $(-1)^n$  a označme  $b_i = \frac{a_i}{(-1)^n}$ . Po převedení  $A^n$  na druhou stranu rovnosti dostaneme

$$b_{n-1} A^{n-1} + \dots + b_1 A + b_0 I = -A^n. \quad (3.14)$$

Rovnost (3.14) nyní přenásobíme zprava libovolným vektorem  $y$ , který však není nulový vektor ani vlastní vektor matice  $A$ , a seřadíme členy na levé straně vzestupně podle exponentu u  $A$ :

$$b_0 Iy + b_1 Ay + \dots + b_{n-1} A^{n-1} y = -A^n y. \quad (3.15)$$

Pokud levou stranu rovnosti zapíšeme maticově, získáme soustavu rovnic s neznámými  $b_0$  až  $b_{n-1}$

$$\left( \begin{array}{c|c|c|c} Iy & Ay & \dots & A^{n-1}y \end{array} \right) \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_{n-1} \end{pmatrix} = -A^n y. \quad (3.16)$$

Sloupce matice této soustavy jsou báze vektory tzv. Krylovova podprostoru  $\mathfrak{K}_n(A, v)$ , jehož využívají i jiné metody hledání vlastních čísel, např. Arnoldiho metoda (více informací v [1]). Tuto soustavu lineárních rovnic lze řešit např. Gaussovou eliminací nebo, pokud si vystačíme s přibližným řešením, pomocí vhodné numerické metody (např. Gaussovy-Seidelovy iterační metody, viz. [8]). Tento způsob výpočtu koeficientů charakteristického polynomu je pro rozměrnější matice mnohem efektivnější než výpočet pomocí determinantu.

Je třeba ještě poznamenat, proč nemůže být vektorem  $y$  nulový vektor nebo vlastní vektor matice  $A$ . Kdyby byl vektor  $y$  nulový, matice soustavy by byla také nulová a nebylo by možné z ní určit koeficienty  $b_i$ . V případě vlastního vektoru by byla dimenze Krylovova podprostoru pouze 1, jelikož jednotlivé vektory  $A^i y$  by byly násobky vektoru  $y$ . Matice soustavy by tak měla hodnotu rovnou 1, a tedy nekonečně mnoho řešení, což znemožňuje jednoznačné nalezení koeficientů  $b_i$ . Naštěstí není složité ověřit, zda náhodně zvolený vektor je vlastním vektorem matice  $A$  (stačí jej přenásobit zleva danou maticí). Teoreticky existuje ještě jeden problém. Pokud zvolíme počáteční vektor tak, že má nulovou složku v podprostoru daném některým z vlastních vektorů, dostaneme opět matici s nižší hodnotí.

Pokud však volíme počáteční vektor jako náhodný, je prakticky nulová pravděpodobnost, že tento problém nastane.

Ukažme si nalezení charakteristického polynomu na jednoduchém příkladu.

### Příklad 3.1

Nalezněme charakteristický polynom matice

$$A = \begin{pmatrix} 1 & 1 & 2 \\ 1 & 2 & 1 \\ 2 & 1 & 1 \end{pmatrix}.$$

Zvolme si nejprve počáteční vektor  $y$ :

$$y = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix},$$

potom sloupce matice soustavy vypadají takto:

$$s_1 = y = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, s_2 = Ay = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}, s_3 = A^2y = \begin{pmatrix} 6 \\ 5 \\ 5 \end{pmatrix}.$$

Pravá strana soustavy:

$$-A^3y = \begin{pmatrix} -21 \\ -21 \\ -22 \end{pmatrix}.$$

Výsledná soustava lineárních rovnic a její řešení:

$$\begin{pmatrix} 1 & 1 & 6 \\ 0 & 1 & 5 \\ 0 & 2 & 5 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} -21 \\ -21 \\ -22 \end{pmatrix} \Rightarrow b = \begin{pmatrix} 4 \\ -1 \\ -4 \end{pmatrix}.$$

Po úpravě  $a_i = b_i(-1)^n$  dostáváme charakteristický polynom

$$p(\lambda) = -\lambda^3 + 4\lambda^2 + \lambda - 4.$$

■

### 3.1.2 Kořeny charakteristického polynomu

Výpočet charakteristického polynomu byl pouze prvním krokem k nalezení vlastních čísel matice  $A$ . Zbývá nám ještě určit kořeny tohoto polynomu. Zde máme dvě možnosti. Buď nalezneme přesné řešení analyticky, anebo použijeme vhodnou numerickou metodu. Jak již bylo řečeno v úvodu této kapitoly, analyticky jsme schopni nalézt kořeny polynomu stupně menšího než 5. U rozměrnějších matic nám poslouží metody numerické. Jednou z těchto metod je metoda půlení intervalů (neboli bisekce), která slouží

k nalezení reálných kořenů reálných polynomů. Jelikož se budeme zabývat převážně maticemi hermitovskými nebo reálnými symetrickými, které mají reálná vlastní čísla, tato metoda pro nás bude postačující.

Metoda půlení intervalů vychází z Darbouxovy vlastnosti spojitě funkce (viz. [7]):

**Věta 3.2** *Nechť funkce  $f : \mathbb{R} \rightarrow \mathbb{R}$  je spojitá na intervalu  $\langle a, b \rangle$ .*

- *Jestliže  $f(a)f(b) < 0$ , potom existuje  $\xi \in (a, b)$  takové, že  $f(\xi) = 0$ .*

Algoritmus metody půlení intervalů začíná na zadaném intervalu  $\langle a_0, b_0 \rangle = \langle a, b \rangle$ , kde platí, že  $f(a)f(b) < 0$ , a postupně vytváří posloupnost intervalů

$$\langle a_0, b_0 \rangle \supset \langle a_1, b_1 \rangle \supset \langle a_2, b_2 \rangle \supset \dots \supset \langle a_n, b_n \rangle.$$

Tyto intervaly vždy obsahují kořen polynomu nacházející se na intervalu  $\langle a, b \rangle$  a délka každého intervalu je rovna polovině délky intervalu předchozího.

Algoritmus této metody lze zapsat takto

```
a;
b;
dokud rozdíl a a b nebude menší, než zadaná přesnost
{
    s = (a+b)/2
    pokud(f(s) má opačné znaménko než f(a))
    {
        a = s;
        b = s;
    }
    pokud(f(s) má opačné znaménko než f(b))
    {
        a = s;
        b = s;
    }
    pokud(f(s) = 0) //byl nalezen kořen
    {
        a = s;
        b = s;
    }
}
```

Jistým problémem této metody je skutečnost, že není schopna najít všechny kořeny na zadaném intervalu, ale pouze jeden. Věta 3.2 garantuje pouze existenci kořenu na zadaném intervalu, ale jejich počet není nijak omezen. Před započítím samotného algoritmu bisekce je tedy nutné provést tzv. separaci kořenů, čili nalézt takové intervaly, aby každý z nich obsahoval právě jeden kořen.

### 3.1.2.1 Separace kořenů

Nejprve musíme nějak vymezit interval, ve kterém budeme kořeny hledat, jelikož definiční obor polynomicke funkce je celá množina reálných čísel. K tomu nám může posloužit např. mocninná metoda, pomocí které nalezneme dominantní vlastní číslo  $\lambda_1$  (jehož absolutní hodnota je spektrální poloměr  $\rho(A)$ ), takže jsme schopni vymezit spektrum matice  $A$ . Ostatní kořeny charakteristického polynomu budou ležet na intervalu  $\langle -\rho(A), \rho(A) \rangle$ . Tento interval nyní rozdělíme na  $m \in \mathbb{N}$  stejně velkých intervalů a vybereme z nich takové intervaly, jejichž krajní body splňují podmínku  $f(a)f(b) < 0$  (výjimečně se může stát, že tímto způsobem nalezneme přímo kořen). Na tyto intervaly aplikujeme již zmíněný algoritmus bisekce. Čím větší  $m$  zvolíme, tím zvýšíme pravděpodobnost toho, že v každém intervalu bude ležet právě jeden kořen.

Tímto způsobem můžeme nalézt pouze kořeny liché násobnosti, jelikož u kořenů sudé násobnosti graf polynomicke funkce neprotíná v daném bodě osu  $x$ , ale pouze se jí dotýká v jednom bodě. Zde nám může pomoci první derivace této polynomicke funkce  $p'(\lambda)$ . Na tuto derivaci aplikujeme výše popsaný algoritmus bisekce. Zde nám již nevadí, že některé kořeny nenajdeme, poněvadž nás nyní zajímají body, kde má polynom  $p(\lambda)$  kořeny sudé násobnosti. V těchto bodech graf  $p'(\lambda)$  protíná osu  $x$ . Nalezneme-li body, ve kterých je  $p'(\lambda) = 0$ , jednoduchým dosazením do  $p(\lambda)$  zjistíme, zda jsou kořeny.

### 3.1.2.2 Násobnost kořenů

Ačkoliv jsme pomocí zde popsaného způsobu našli vlastní čísla matice  $A$ , neznáme jejich algebraické násobnosti. Zde nám nezbyvá nic jiného, než vydělit polynom  $p(\lambda)$  kořenovým činitelem  $(\lambda - \lambda_i)^k$ . Vzhledem k tomu, že jsme našli pouze přibližnou hodnotu kořene, bude mít tento podíl nenulový zbytek. Pokud však bude tento zbytek velmi blízký nule (se zadanou přesností), bude mít kořen  $\lambda_i$  násobnost  $k$ .

### 3.1.2.3 Shrnutí

Vidíme, že hledání vlastních čísel pomocí charakteristického polynomu není příliš vhodná metoda. Existuje zde totiž riziko, že nenalezneme všechna vlastní čísla, zvláště v případech, kdy jsou jejich hodnoty navzájem blízké, nebo nalezneme vlastní číslo, které ve skutečnosti vlastním číslem není, a to v místech, kde se graf  $p(\lambda)$  přiblíží k ose  $x$  na menší vzdálenost, než byla námi požadovaná přesnost výpočtu. Proto se v praxi používají spíše jiné numerické metody hledání vlastních čísel.

## 3.2 Mocninná metoda

Mocninná metoda je jednoduchá iterační metoda určená k výpočtu vlastních čísel a vektorů. Vytváří postupně posloupnost vektorů  $A^k v$ , kde  $v$  je počáteční vektor, pro jehož volbu platí určité podmínky. Tato posloupnost za jistých podmínek konverguje k vlastnímu vektoru, který náleží dominantnímu vlastnímu číslu  $\lambda_1$ . Vzhledem k tomu, že norma vektoru může postupně narůstat, je vhodné v každém kroku získaný vektor

normalizovat. Předpis pro mocninovou metodu vypadá tedy následovně:

$$v_k = \frac{1}{\nu_k} A v_{k-1}, \quad (3.17)$$

kde  $\nu_k$  je norma vektoru  $A v_{k-1}$  a  $k$  udává počet vykonaných iterací (ten je dán požadovanou přesností výpočtu). Konvergenci mocninné metody lze shrnout v následující větě:

**Věta 3.3** *Nechť  $A$  je komplexní čtvercová matice dimenze  $n$ , jejíž vlastní vektory tvoří bázi vektorového prostoru dimenze  $n$  a jež má pouze jedno dominantní vlastní číslo, jehož algebraická násobnost je rovna maximálnímu počtu lineárně nezávislých vlastních vektorů, které náleží tomuto vlastnímu číslu. Potom pro libovolný počáteční vektor  $v_0$ , jež není vlastním vektorem matice  $A$  a zároveň má nenulovou složku v invariantním podprostoru příslušném vlastnímu číslu  $\lambda_1$ , platí, že posloupnost (3.17) postupně konverguje k vlastnímu vektoru  $v$ , který náleží vlastnímu číslu  $\lambda_1$ , a posloupnost  $\nu_k$  postupně konverguje k dominantnímu vlastnímu číslu.*

**Důkaz.** Jelikož matice  $A$  má právě  $n$  lineárně nezávislých vlastních vektorů, můžeme počáteční vektor  $v_0$ , zapsat jako jejich lineární kombinaci

$$v_0 = a_1 v_1 + a_2 v_2 + \dots + a_n v_n, \quad (3.18)$$

kde  $a_i$  jsou skaláry a  $v_i$  vlastní vektory matice  $A$ . Z předpokladu věty musí být  $a_1 \neq 0$  (nenulovost složky v invariantním podprostoru příslušném vlastnímu číslu  $\lambda_1$ ) a alespoň jeden koeficient  $a_i \neq 0$ ,  $i \in \{2, 3, \dots, n\}$ . Potom můžeme nad touto rovností provést následující ekvivalentní úpravy:

$$A v_0 = a_1 A v_1 + a_2 A v_2 + \dots + a_n A v_n \quad (3.19)$$

$$A v_0 = a_1 \lambda_1 v_1 + a_2 \lambda_2 v_2 + \dots + a_n \lambda_n v_n \quad (3.20)$$

$$A^2 v_0 = a_1 \lambda_1^2 v_1 + a_2 \lambda_2^2 v_2 + \dots + a_n \lambda_n^2 v_n \quad (3.21)$$

$$\vdots$$

$$A^k v_0 = a_1 \lambda_1^k v_1 + a_2 \lambda_2^k v_2 + \dots + a_n \lambda_n^k v_n \quad (3.22)$$

$$A^k v_0 = \lambda_1^k \left( a_1 v_1 + a_2 \left( \frac{\lambda_2}{\lambda_1} \right)^k v_2 + \dots + a_n \left( \frac{\lambda_n}{\lambda_1} \right)^k v_n \right) \quad (3.23)$$

Pro vlastní čísla  $\lambda_i$  platí  $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$ . Proto v poslední rovnosti budou členy  $\left( \frac{\lambda_n}{\lambda_1} \right)^k$  konvergovat k 0. Pro dostatečně velké  $k$  můžeme tedy psát

$$A^k v_0 = \lambda_1^k a_1 v_1. \quad (3.24)$$

Posloupnost skutečně konverguje ke skalárnímu násobku vlastního vektoru  $v_1$ . Navíc při normování v každém kroku posloupnost konverguje k normovanému vlastnímu vektoru. Skutečnost, že posloupnost  $\nu_k$  konverguje k dominantnímu vlastnímu číslu, vyplývá z rovnosti (3.17), jelikož ji lze jednoduše upravit do tvaru

$$\nu_k v_k = A v_{k-1}. \quad (3.25)$$

**Poznámka 3.1** Z důkazu lze vyčíst i rychlost konvergence mocninné metody. Ta je nepřímo úměrná absolutní hodnotě podílu

$$\left| \frac{\lambda_2}{\lambda_1} \right|.$$

Je-li tedy v absolutní hodnotě druhé největší vlastní číslo blíže spektrálnímu poloměru, pak bude mocninná metoda konvergovat pomaleji.

Jestliže má matice  $A$  dominantní vlastní číslo větší násobnosti než 1, bude posloupnost (3.17) konvergovat do invariantního podprostoru příslušného k  $\lambda_1$ . Násobnost takového vlastního čísla pomocí mocninné metody nezjistíme. Pokud však zvolíme jiný počáteční vektor  $v_0$ , dojdeme s nejvyšší pravděpodobností k jinému vektoru, který opět leží v daném invariantním podprostoru. Je-li dominantní vlastní číslo násobnosti 1, bude algoritmus konvergovat opět ke stejnému vektoru (nebo k vektoru opačnému). V případě, že matice  $A$  má více dominantních vlastních čísel (např. 5 a  $-5$ ), nebude mocninná metoda konvergovat.

Zdrojový kód algoritmu mocninné metody lze nalézt v příloze, výpis zdrojového kódu č. 3.

V praxi nám obvykle nestačí nalézt pouze dominantní vlastní číslo. Proto byly vyvinuty různé modifikace mocninné metody k nalezení ostatních vlastních čísel.

### 3.2.1 Posun spektra

Tato modifikace se používá převážně v případech, kdy mocninná metoda nekonverguje z důvodu většího počtu dominantních vlastních čísel. Vychází z následující věty:

**Věta 3.4** *Nechť  $A$  je komplexní čtvercová matice dimenze  $n$  s vlastními čísly  $\lambda_i$ ,  $i = \{1, 2, \dots, n\}$ . Potom vlastní čísla matice  $A' = A + \sigma I$ ,  $\sigma \in \mathbb{R}$  jsou rovny  $\lambda'_i = \lambda_i + \sigma$ . Vlastní vektory zůstávají zachovány.*

**Důkaz.** Při důkazu vyjdeme z rovnosti (2.1), ke které přičteme na obou stranách  $\sigma$  násobek vlastního vektoru  $v$ :

$$Av + I\sigma v = \lambda v + \sigma v. \quad (3.26)$$

Tuto rovnost jednoduše upravíme do tvaru

$$(A + \sigma I)v = (\lambda + \sigma)v, \quad (3.27)$$

a dostáváme tak vztah pro vlastní čísla a vlastní vektory matice  $A + \sigma I$ , čímž jsme přímo dokázali větu 3.4. ■

Iterační předpis mocninné metody s posunem spektra vypadá následovně:

$$v_k = \frac{1}{\nu_k}(A + \sigma I)v_{k-1}. \quad (3.28)$$

Dostáváme tak vlastní vektor matice  $A$ . Příslušné vlastní číslo zjistíme odečtením  $\sigma$  od získaného  $\lambda$ . Posunutí spektra nemusí být užitečné pouze v případě více dominantních vlastních čísel. Pokud použitím mocninné metody vypočteme dominantní vlastní číslo, posunem spektra  $(A - \lambda_1 I)$  můžeme nalézt vlastní číslo na opačné straně spektra, tedy vlastní číslo, které je nejbližší  $-\lambda_1$ . V případě, že je matice  $A$  symetrická pozitivně definitní, dostaneme tímto posunem v absolutní hodnotě nejmenší vlastní číslo.

Dále lze vhodnou volbou  $\sigma$  zvýšit (nebo naopak snížit) rychlost konvergence mocninné metody, jelikož posunem dochází ke změně poměru

$$\left| \frac{\lambda_2}{\lambda_1} \right|.$$

Zdrojový kód algoritmu mocninné metody s posunem spektra lze nalézt v příloze, výpis zdrojového kódu č. 4.

### 3.2.2 Inverzní matice

Jestliže potřebujeme najít v absolutní hodnotě nejmenší vlastní číslo, je vhodné aplikovat mocninnou metodu s inverzní maticí. Pro vlastní čísla inverzní matice platí následující tvrzení:

**Věta 3.5** *Nechť  $A$  je komplexní invertovatelná čtvercová matice dimenze  $n$  s nenulovými vlastními čísly  $\lambda_i$ ,  $i = \{1, 2, \dots, n\}$ . Potom vlastní čísla matice  $A' = A^{-1}$  jsou rovny  $\lambda'_i = \frac{1}{\lambda_i}$ . Vlastní vektory zůstávají zachovány.*

**Důkaz.** Při důkazu opět vyjdeme z rovnosti (2.1), nad kterou provedeme následující ekvivalentní úpravy:

$$Av = \lambda v \quad / A^{-1} \cdot \quad (3.29)$$

$$A^{-1}Av = \lambda A^{-1}v \quad (3.30)$$

$$v = \lambda A^{-1}v / \cdot \frac{1}{\lambda} \quad (3.31)$$

$$\frac{1}{\lambda}v = A^{-1}v \quad (3.32)$$

Došli jsme k rovnosti, která přímo dokazuje naše tvrzení. ■

Nahradíme-li v algoritmu mocninné metody matici  $A$  maticí  $A^{-1}$ , bude algoritmus konvergovat k dominantnímu vlastnímu číslu matice  $A^{-1}$ , jehož převrácená hodnota je rovna v absolutní hodnotě nejmenšímu vlastnímu číslu matice  $A$ .

Nabízí se možnost zkombinovat posun a inverzi spektra. Jestliže nejprve posuneme spektrum matice  $A$  o hodnotu  $-\sigma$ , poté provedeme nad touto maticí inverzi, dostaneme iterační předpis

$$v_k = \frac{1}{\nu_k} (A - \sigma I)^{-1} v_{k-1}. \quad (3.33)$$

Algoritmus bude konvergovat k dominantnímu vlastnímu vektoru matice  $(A - \sigma I)^{-1}$ . Jelikož jsme provedli pouze operace posunu a inverze, zůstávají vlastní vektory zachovány. Odpovídající vlastní číslo matice  $A$  získáme zpětnou substitucí

$$\lambda = \frac{1}{\lambda' - \sigma}. \quad (3.34)$$

Tuto kombinaci posunu jsme zvolili, protože jsme s její pomocí schopni nalézt vlastní číslo, které je nejbližší zadané hodnotě  $\sigma$ , takže již disponujeme aparátem k nalezení libovolného vlastního čísla a vlastního vektoru matice  $A$ .

Algoritmus nalezení vlastního čísla nejbližší zadané hodnotě lze nalézt v příloze, výpis zdrojového kódu č. 5.

Největším problémem této modifikace mocinné metody je nalezení inverze, což je zpravidla výpočetně velmi náročná operace. Existuje několik způsobů, jak ji nalézt.

### 3.2.2.1 LU rozklad

K nalezení inverze je vhodné použít LU rozkladu (viz. [5]). Postup výpočtu je následující:

1. Nalezneme LU rozklad matice  $A - \sigma I = LU$
2. Opakujeme tento postup, dokud nedosáhneme požadované přesnosti:

$$v_k = \frac{1}{\nu_k} U^{-1} L^{-1} v_{k-1}.$$

3. Dostaneme vlastní vektor  $v$ . Příslušné vlastní číslo získáme užitím vztahu (3.34).

### 3.2.2.2 Cayleyho-Hamiltonova věta

Již v části 3.1.2 jsme si uvedli Cayleyho-Hamiltonovu větu. Této věty nyní využijeme k odvození předpisu pro výpočet inverzní matice. Využijeme také skutečnosti, že absolutní člen charakteristického polynomu je roven determinantu matice (viz. kapitola 1). Provedeme následující ekvivalentní úpravy:

$$a_n A^n + a_{n-1} A^{n-1} + \dots + a_1 A + a_0 I = O \quad (3.35)$$

$$a_n A^n + a_{n-1} A^{n-1} + \dots + a_1 A + \det(A) I = O \quad (3.36)$$

$$a_n A^n + a_{n-1} A^{n-1} + \dots + a_1 A = -\det(A) I \quad (3.37)$$

$$A(a_n A^{n-1} + a_{n-1} A^{n-2} + \dots + a_1 I) = -\det(A) I / A^{-1}. \quad (3.38)$$

$$(a_n A^{n-1} + a_{n-1} A^{n-2} + \dots + a_1 I) = A^{-1}(-\det(A)) \quad (3.39)$$

$$-\frac{1}{\det(A)}(a_n A^{n-1} + a_{n-1} A^{n-2} + \dots + a_1 I) = A^{-1} \quad (3.40)$$

Máme tedy předpis pro nalezení inverzní matice. Charakteristický polynom určíme např. užitím Krylovovy metody. Tento předpis je vhodný, pokud potřebujeme nalézt pouze jedno vlastní číslo. Pokud bychom chtěli nalézt více vlastních čísel, museli bychom pro každé číslo  $\sigma$  hledat charakteristický polynom zvlášť, což je při použití Krylovovy metody



neefektivní. Odvodíme zde způsob, jak nalézt koeficienty charakteristického polynomu matice  $A - \sigma I$  pouze při znalosti koeficientů charakteristického polynomu matice  $A$ . Označme  $A'$  matici s posunutým spektrem vůči matici  $A$ ,  $\lambda'$  vlastní číslo matice  $A'$  a  $\lambda$  jemu odpovídající vlastní číslo matice  $A$ , neboli

$$A' = A - \sigma I, \quad (3.41)$$

$$\lambda' = \lambda - \sigma. \quad (3.42)$$

Těmto maticím odpovídají charakteristické polynomy

$$p_A(\lambda) = a_n \lambda^n + a_{n-1} \lambda^{n-1} + \dots + a_1 \lambda + a_0 \quad (3.43)$$

$$p_{A'}(\lambda') = b_n (\lambda')^n + b_{n-1} (\lambda')^{n-1} + \dots + b_1 \lambda' + b_0. \quad (3.44)$$

S využitím rovnosti (3.42) můžeme druhý z těchto polynomů zapsat také takto:

$$p_{A'}(\lambda') = p_A(\lambda' + \sigma) = a_n (\lambda' + \sigma)^n + a_{n-1} (\lambda' + \sigma)^{n-1} + \dots + a_1 (\lambda' + \sigma) + a_0. \quad (3.45)$$

Na dvojčleny  $(\lambda' + \sigma)$  aplikujeme binomickou větu:

$$p_{A'}(\lambda') = a_n \sum_{k=0}^n \binom{n}{k} (\lambda')^{n-k} \sigma^k + a_{n-1} \sum_{k=0}^{n-1} \binom{n-1}{k} (\lambda')^{n-1-k} \sigma^k + \dots + a_1 \lambda' + a_1 \sigma + a_0. \quad (3.46)$$

Seskupením členů se stejným exponentem u  $\lambda'$  a vytknutím  $(\lambda')^i$  z každé z těchto skupin dostaneme vztahy pro výpočet koeficientů  $b_i$ . Maticově je lze zapsat takto:

$$b = T a, \quad (3.47)$$

kde

$$b = \begin{pmatrix} b_n \\ b_{n-1} \\ \vdots \\ b_0 \end{pmatrix} \quad a = \begin{pmatrix} a_n \\ a_{n-1} \\ \vdots \\ a_0 \end{pmatrix}$$

a

$$T = \begin{pmatrix} \binom{n}{0} \sigma^0 & 0 & 0 & \dots & 0 & 0 \\ \binom{n}{1} \sigma^1 & \binom{n-1}{0} \sigma^0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & & \vdots & \vdots \\ \vdots & \vdots & & \ddots & \vdots & \vdots \\ \binom{n}{n-1} \sigma^{n-1} & \binom{n-1}{n-2} \sigma^{n-2} & \dots & \dots & \binom{1}{0} \sigma^0 & 0 \\ \binom{n}{n} \sigma^n & \binom{n-1}{n-1} \sigma^{n-1} & \dots & \dots & \binom{1}{1} \sigma^1 & \binom{0}{0} \sigma^0 \end{pmatrix}.$$

Stačí tedy sestavit matici  $T$  pro posun  $\sigma$ . To lze provést např. užitím následujícího vztahu:

$$T = \sum_{i=0}^n \frac{1}{i!} T_0^i, \quad (3.48)$$

kde

$$T_0 = \begin{pmatrix} 0 & 0 & 0 & \dots & \dots & 0 \\ n\sigma & 0 & 0 & \dots & \dots & 0 \\ 0 & (n-1)\sigma & 0 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 & 0 \\ 0 & \dots & \dots & 0 & 1\sigma & 0 \end{pmatrix}. \quad (3.49)$$

Dále platí rovnost

$$\det(A') = \det(A - \sigma I) = p_A(\sigma). \quad (3.50)$$

Inverzi matice  $A$  s posunutým spektrem najdeme tedy takto:

$$(A')^{-1} = (A - \sigma I)^{-1} = -\frac{1}{p_A(\sigma)}(b_n(A')^{n-1} + b_{n-1}(A')^{n-2} + \dots + b_1 I). \quad (3.51)$$

Koeficienty  $b_i$  určíme výše odvozeným způsobem.

### 3.3 Jacobiho metoda

Jacobiho metoda je iterační metoda určena pro hledání vlastních čísel a vektorů reálných symetrických matic. Metoda využívá Givensovy matice rovinné rotace ve tvaru:

$$G_{i,j} = \begin{pmatrix} 1 & \dots & 0 & \dots & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \dots & \cos \phi & \dots & \sin \phi & \dots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \dots & -\sin \phi & \dots & \cos \phi & \dots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \dots & 0 & \dots & 1 \end{pmatrix}. \quad (3.52)$$

Goniometrické funkce se vyskytují na řádcích a sloupcích s indexy  $i$  a  $j$ , kde  $i < j$ . Givensova matice se využívá k vynulování prvku v  $i$ -tém řádku a  $j$ -tém sloupci reálné symetrické matice  $S$ . Jacobiho metoda v každém svém kroku vypočítá matici  $G$  k matici  $S$ , a to tak, aby po přenásobení  $S' = G^T S G$  došlo k vynulování v absolutní hodnotě největšího mimodiagonálního prvku (v dalším výkladu jej budeme nazývat pivot). Jeli-kož  $G$  je ortogonální matice, zachovávají se při této transformaci vlastní čísla (jedná se o podobnostní transformaci). Opakováním tohoto postupu získáme „téměř diagonální“ matici  $D$ , tj. matici s mimodiagonálními prvky blízkými 0. Na diagonále matice  $D$  se tedy nacházejí přibližné hodnoty vlastních čísel matice  $S$ .

Podívejme se, jak v každé iteraci nalézt vhodný úhel  $\phi$ . Nejprve je třeba nalézt souřadnice pivotu  $i$  a  $j$ . Pokud sestavíme potřebnou Givensovu matici  $G$  a provedeme transformaci  $S' = G^T S G$ , budou prvky matice  $S'$  vypadat takto:

$$S'_{ii} = c^2 S_{ii} + 2sc S_{ij} + s^2 S_{jj} \quad (3.53)$$

$$S'_{jj} = c^2 S_{jj} - 2scS_{ij} + s^2 S_{ii} \quad (3.54)$$

$$S'_{ij} = S'_{ji} = (c^2 - s^2)S_{ij} + sc(S_{jj} - S_{ii}) \quad (3.55)$$

$$S'_{ik} = S'_{ki} = cS_{ik} + sS_{jk} \quad k \neq i, j \quad (3.56)$$

$$S'_{jk} = S'_{kj} = cS_{jk} - sS_{ik} \quad k \neq i, j \quad (3.57)$$

$$S'_{kl} = S_{kl} \quad k, l \neq i, j \quad (3.58)$$

Konstanta  $c$  zde označuje  $\cos \phi$  a  $s$  označuje  $\sin \phi$ . My se nyní zaměříme na rovnici (3.55). Cílem každého kroku Jacobiho metody je vynulování pivotu a právě vztah (3.55) vyjadřuje hodnotu pivotu po transformaci. Získáváme tedy rovnici

$$(\cos^2 \phi - \sin^2 \phi)S_{ij} + \sin \phi \cos \phi (S_{jj} - S_{ii}) = 0 \quad (3.59)$$

Pokud  $S_{jj} \neq S_{ii}$ , pak jednoduchou úpravou s využitím vztahů

$$\cos^2 \phi - \sin^2 \phi = \cos(2\phi), \quad (3.60)$$

$$2 \sin \phi \cos \phi = \sin(2\phi) \quad (3.61)$$

dostáváme

$$\tan(2\phi) = \frac{2S_{ij}}{S_{ii} - S_{jj}} \quad (3.62)$$

Odtud již snadno určíme úhel  $\phi$ :

$$\phi = \frac{1}{2} \arctan \frac{2S_{ij}}{S_{ii} - S_{jj}} \quad (3.63)$$

V případě, že  $S_{jj} = S_{ii}$ , přechází vztah (3.55) v rovnici

$$(\cos^2 \phi - \sin^2 \phi)S_{ij} = 0 \quad (3.64)$$

Zde opět využijeme vztahu (3.60), a dostáváme tak rovnici:

$$\cos(2\phi)S_{ij} = 0 \quad (3.65)$$

Řešením této rovnice je např.  $\phi = \frac{\pi}{4}$ . Máme tedy vztah pro nalezení úhlu  $\phi$  potřebného k sestavení Givensovy matice  $G$ .

Zbývá nám ještě dokázat následující větu o konvergenci Jacobiho metody:

**Věta 3.6** *Nechť  $A$  je symterická matice. Potom algoritmus Jacobiho metody postupně konverguje k diagonální matici  $D$ .*

**Důkaz.** Označme  $\Gamma(S)$  součet druhých mocnin všech mimodiagonálních prvků matice  $S$ , neboli

$$\Gamma(S) = \sum_{i,j=1, i \neq j}^n S_{ij}^2 \quad (3.66)$$

Naším cílem je nalézt vztah mezi  $\Gamma(S)$  a  $\Gamma(S')$ . K tomu nám pomohou rovnice (3.53) až (3.58). Z (3.58) vyplývá, že  $(S'_{kl})^2 = S_{kl}^2$  pro  $k, l \neq i, j$ . Nyní se zaměříme na sloupce s indexy  $i$  a  $j$ . Vezmeme pevný index  $k \neq i, j$  a podíváme se, jak se chová součet  $(S'_{ki})^2 + (S'_{kj})^2$ :

$$\begin{aligned} (S'_{ki})^2 + (S'_{kj})^2 &= (\cos \phi S_{ki} + \sin \phi S_{kj})^2 + (\cos \phi S_{kj} - \sin \phi S_{ki})^2 = \\ &= \cos^2 \phi S_{ki}^2 + 2 \cos \phi \sin \phi S_{ki} S_{kj} + \sin^2 \phi S_{kj}^2 + \\ &+ \cos^2 \phi S_{kj}^2 - 2 \cos \phi \sin \phi S_{ki} S_{kj} + \sin^2 \phi S_{ki}^2 = \\ &= (\cos^2 \phi + \sin^2 \phi) S_{ki}^2 + (\cos^2 \phi + \sin^2 \phi) S_{kj}^2 = S_{ki}^2 + S_{kj}^2 \end{aligned}$$

Vidíme, že součet  $(S'_{ki})^2 + (S'_{kj})^2$  přispívá do  $\Gamma(S')$  stejnou hodnotou, jako  $S_{ki}^2 + S_{kj}^2$  do  $\Gamma(S)$ . Stejná vlastnost platí také pro řádky s indexy  $i$  a  $j$ , takže  $(S'_{ik})^2 + (S'_{jk})^2 = S_{ik}^2 + S_{jk}^2$ . Zbývá nám ještě pivot, tedy  $S_{ij} = S_{ji}$ . U něj však víme, že  $(S'_{ij})^2 = (S'_{ji})^2 = 0$ . Nyní již není obtížné odvodit hledaný vztah:

$$\Gamma(S') = \Gamma(S) - 2S_{ij}^2 \quad (3.67)$$

Pokud dokážeme, že posloupnost  $\Gamma(S')$  konverguje k 0, znamená to, že  $S'$  postupně konverguje k  $D$ . Podíváme se na nejhorší případ, a to tehdy, pokud jsou mimodiagonální prvky svou hodnotou rovny hodnotě pivotu. Pokud označíme počet mimodiagonálních prvků  $2N = n(n-1)$ , bude v tomto případě platit:

$$\Gamma(S') = \Gamma(S) \left(1 - \frac{1}{N}\right) \quad (3.68)$$

Obecně tedy, pro libovolné hodnoty mimodiagonálních prvků, platí, že

$$0 \leq \Gamma(S') \leq \Gamma(S) \left(1 - \frac{1}{N}\right) \quad (3.69)$$

Označme  $\Gamma(S')^{(k)}$  součet druhých mocnin všech mimodiagonálních prvků matice  $S'$  po provedení  $k$  iterací. Dále označme  $S_0$  původní reálnou symetrickou matici. Pak platí

$$0 \leq \Gamma(S')^{(k)} \leq \Gamma(S_0) \left(1 - \frac{1}{N}\right)^k \quad (3.70)$$

Jelikož  $N \geq 1, N \in \mathbb{N}$ , je

$$\lim_{k \rightarrow \infty} \left(1 - \frac{1}{N}\right)^k = 0, \quad (3.71)$$

takže  $\Gamma(S')$  postupně konverguje k 0, což jsme chtěli dokázat. ■

Kromě vlastních čísel je třeba vypočítat i vlastní vektory matice  $S$ . Zde stačí vynásobit všechny Givensovy matice, které jsme určili v každé iteraci, tedy

$$V = G_1 G_2 \dots G_k. \quad (3.72)$$

Sloupce matice  $V$  jsou aproximace vlastních vektorů zadané matice  $S$ ,  $k$  udává počet vykonaných iterací Jacobiho algoritmu.

Výsledný Jacobiho algoritmus lze zapsat následovně:

```
S = A;
Q = I;
dokud S nesplní požadovanou přesnost
{
    najdi pivot matice S;
    sestav matici G;
    S = G' S G; //G' značí transponovanou matici G
    Q = Q G;
}
```

Zdrojový kód v Matlabu lze nalézt v příloze, výpis zdrojového kódu č. 1.

Podívejme se nyní na konkrétní příklad výpočtu vlastních čísel a vlastních vektorů užitím Jacobiho metody.

### Příklad 3.2

Pomocí Jacobiho metody spočteme vlastní čísla a vektory matice

$$S = \begin{pmatrix} 5 & 3 & 7 \\ 3 & 5 & 7 \\ 7 & 7 & 15 \end{pmatrix}$$

V první iteraci bude jako pivot zvolen prvek  $S_{13}$ . Takto budou vypadat Givensova matice  $G$ , matice  $S$  po transformaci (budoucí diagonální matice) a matice  $V$ , která bude obsahovat vlastní vektory:

$$G_1 = \begin{pmatrix} 0,889 & 0,000 & 0,458 \\ 0,000 & 1,000 & 0,000 \\ -0,458 & 0,000 & 0,889 \end{pmatrix} S_1 = \begin{pmatrix} 1,398 & -0,536 & 0,000 \\ -0,536 & 5,000 & 7,597 \\ -0,458 & 7,597 & 18,602 \end{pmatrix}$$

$$V_1 = \begin{pmatrix} 0,889 & 0,000 & 0,458 \\ 0,000 & 1,000 & 0,000 \\ -0,458 & 0,000 & 0,889 \end{pmatrix}$$

Ve druhé iteraci bude pivotem prvek  $S_{23}$ :

$$G_2 = \begin{pmatrix} 1,000 & 0,000 & 0,000 \\ 0,000 & 0,913 & 0,408 \\ 0,000 & -0,408 & 0,913 \end{pmatrix} S_2 = \begin{pmatrix} 1,398 & -0,489 & -0,219 \\ -0,489 & 1,605 & 0,000 \\ -0,219 & 0,000 & 21,998 \end{pmatrix}$$

$$V_2 = \begin{pmatrix} 0,889 & -0,187 & 0,418 \\ 0,000 & 0,913 & 0,408 \\ -0,458 & -0,363 & 0,812 \end{pmatrix}$$

Ve třetí iteraci bude pivotem prvek  $S_{12}$ . Za povšimnutí stojí jeho absolutní hodnota (již po dvou iteracích je menší než 0,5). Podívejme se, jak budou matice  $S$  a  $V$  vypadat po pěti iteracích:

$$S_5 = \begin{pmatrix} 1,000 & 0,001 & 0,000 \\ 0,001 & 2,000 & 0,000 \\ 0,000 & 0,000 & 22,000 \end{pmatrix} \quad V_5 = \begin{pmatrix} 0,577 & -0,708 & 0,408 \\ 0,578 & 0,706 & 0,408 \\ -0,577 & 0,001 & 0,816 \end{pmatrix}$$

Již po pěti iteracích máme velmi dobrou aproximaci. Pro srovnání zde uvádím vlastní čísla a normované vlastní vektory vypočítané analyticky:

$$\begin{aligned} \lambda_1 = 1 \quad \lambda_2 = 2 \quad \lambda_3 = 22 \\ v_1 = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix} \doteq \begin{pmatrix} 0,577 \\ 0,577 \\ -0,577 \end{pmatrix} \quad v_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix} \doteq \begin{pmatrix} -0,707 \\ 0,707 \\ 0 \end{pmatrix} \\ v_3 = \frac{1}{\sqrt{6}} \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} \doteq \begin{pmatrix} 0,408 \\ 0,408 \\ 0,816 \end{pmatrix} \end{aligned}$$

■

**Poznámka 3.2** Jacobiho metodu lze zobecnit i pro hermitovské matice. Podobnostní transformace má tvar

$$A' = R_{i,j} A R_{i,j}^H, \quad (3.73)$$

nebo lze problém převést na reálnou symetrickou matici, viz. např. [3].

### 3.4 QR metoda

Stejně jako Jacobiho metoda je QR metoda iterační metoda určena pro hledání vlastních čísel a vektorů reálných symetrických matic. Využívá tzv. QR rozkladu matice

$$A = QR, \quad (3.74)$$

kde  $Q$  je ortogonální matice a  $R$  je horní trojúhelníková matice. Vzhledem k tomu, že matice  $Q$  je ortogonální, má matice

$$A' = Q^T A Q, \quad (3.75)$$

stejně vlastní vektory a stejná vlastní čísla jako matice  $A$ , jedná se tedy o podobnostní transformaci. Opakováním tohoto postupu konverguje matice  $A'$  k diagonální matici  $D$  s vlastními čísly na diagonále (Důkaz konvergence QR algoritmu je velmi netriviální, lze

jej nalézt např. v [2]). Rovnici (3.75) lze za použití vztahu (3.74) jednoduše upravit do tvaru

$$A' = RQ. \quad (3.76)$$

Naskýtá se však otázka, jak nalézt samotný QR rozklad matice  $A \in \mathbb{C}^{n \times n}$ . Existuje několik způsobů hledání QR rozkladu, např. využitím Givensovy nebo Householderovy transformace. Jelikož jsme se Givensově transformaci věnovali v minulé kapitole, podíváme se na transformaci Householderovu. Ta využívá matice rovinného zrcadlení  $P$ . Transformace  $P$  zobrazí vektor  $v$  na vektor  $Pv$ , který má stejnou velikost a je násobkem prvního vektoru standardní báze, tedy

$$v = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \quad Pv = \begin{pmatrix} \|v\| \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad (3.77)$$

Matice  $P$  má tvar

$$P = I - 2 \frac{ww^T}{w^T w}, \quad (3.78)$$

kde  $w = -\operatorname{sgn}(v_1)\|v\|e_1 - v$ ,  $I$  je jednotková matice a  $e_1$  je první vektor standardní báze. Odvození Householderovy matice lze nalézt např. v [6]. Pomocí postupných Householderových projekcí lze matici  $A$  převést na horní trojúhelníkovou matici  $R$ , tedy

$$R = P_{n-1}P_{n-2} \dots P_1 A. \quad (3.79)$$

Vzhledem k tomu, že matice  $P_i, i \in \{1, 2, \dots, n-1\}$  jsou symetrické a ortogonální, lze rovnici (3.79) přepsat do tvaru

$$A = P_1 P_2 \dots P_{n-1} R. \quad (3.80)$$

Odtud lze již vyjádřit matici  $Q$ :

$$Q = P_1 P_2 \dots P_{n-1}. \quad (3.81)$$

Nutno poznamenat, že pokud bychom hledali matice  $P_i$  podle vztahu (3.78), budou mít rozdílné dimenze. Každá další matice  $P$  totiž zobrazuje vektor dimenze o 1 menší než byla dimenze vektoru předchozího (chceme vynulovat pouze prvky pod diagonálou). Abychom mohli matice  $P_i$  mezi sebou násobit, je třeba je rozšířit. Označme tedy  $P'_i$  nalezenou Householderovu matici, potom rozšířená matice vypadá blokově takto:

$$P_i = \left( \begin{array}{c|c} I & O \\ \hline O & P'_i \end{array} \right), \quad (3.82)$$

kde  $I$  je jednotková matice a  $O$  je matice nulová. Tento tvar matice  $P_i$  nám zajišťuje, že se vynulují prvky daného sloupce pod diagonálou, aniž by se změnily prvky ve sloupcích nalevo od tohoto sloupce.

Ukažme si nalezení QR rozkladu pomocí Householderovy transformace na příkladu.

**Příklad 3.3**

Najdeme QR rozklad matice

$$A = \begin{pmatrix} 7 & 6 & 3 \\ 6 & 9 & 4 \\ 3 & 4 & 3 \end{pmatrix}.$$

První transformace nám vynuluje prvky pod diagonálou prvního sloupce:

$$v = \begin{pmatrix} 7,000 \\ 6,000 \\ 3,000 \end{pmatrix} \quad w = \begin{pmatrix} 2,695 \\ -6,000 \\ -3,000 \end{pmatrix} \quad P_1 = \begin{pmatrix} 0,722 & 0,619 & 0,309 \\ 0,618 & -0,378 & -0,689 \\ 0,309 & -0,689 & 0,656 \end{pmatrix}$$

$$P_1 A = \begin{pmatrix} 9,695 & 11,139 & 5,570 \\ 0,000 & -2,440 & -1,720 \\ 0,000 & -1,720 & 0,140 \end{pmatrix}$$

Druhá transformace nám vynuluje prvky pod diagonálou druhého sloupce:

$$v = \begin{pmatrix} -2,440 \\ -1,720 \end{pmatrix} \quad w = \begin{pmatrix} 5,426 \\ 1,720 \end{pmatrix} \quad P'_2 = \begin{pmatrix} -0,817 & -0,576 \\ -0,576 & -0,817 \end{pmatrix}$$

$$P_2 = \begin{pmatrix} 1,000 & 0,000 & 0,000 \\ 0,000 & -0,817 & -0,576 \\ 0,000 & -0,576 & -0,817 \end{pmatrix}$$

$$R = P_2 P_1 A = \begin{pmatrix} 9,695 & 11,139 & 5,570 \\ 0,000 & 2,986 & 1,325 \\ 0,000 & 0,000 & 1,105 \end{pmatrix}$$

Našli jsme tedy matici  $R$ . Zbývá nám ještě matice  $Q$ :

$$Q = P_1 P_2 = \begin{pmatrix} 0,722 & -0,684 & -0,104 \\ 0,619 & 0,706 & -0,345 \\ 0,309 & 0,185 & 0,933 \end{pmatrix}$$

■

Vraťme se nyní k samotnému QR algoritmu. Už víme, že postupným násobením  $RQ$  získáme matici s vlastními čísly na diagonále (resp. s jejich přibližnými hodnotami). My však potřebujeme nalézt také vlastní vektory matice  $A$ . K tomu nám stačí vynásobit mezi sebou všechny získané matice  $Q_i$ ,  $i \in \{1, 2, \dots, k\}$ , kde  $k$  udává počet vykonaných iterací, tedy

$$V = Q_1 Q_2 \dots Q_k. \quad (3.83)$$

Výsledný QR algoritmus můžeme tedy zapsat takto:

$S = A$ ;

$V = I$ ;

dokud  $S$  nesplní požadovanou přesnost



```

{
    najdi QR rozklad matice S;
    S = RQ;
    V = VQ;
}

```

Zdrojový kód v Matlabu lze nalézt v příloze, výpis zdrojového kódu č. 2  
Ilustrujme si QR algoritmus na příkladu.

### Příklad 3.4

Najděme vlastní čísla a vlastní vektory matice

$$A = \begin{pmatrix} 5 & 3 & 7 \\ 3 & 5 & 7 \\ 7 & 7 & 15 \end{pmatrix}.$$

(je to stejná matice, na které byla demonstrována Jacobiho metoda). V první iteraci pomocí Householderovy transformace vypočteme QR rozklad této matice:

$$Q_1 = \begin{pmatrix} -0,549 & 0,630 & -0,550 \\ -0,329 & -0,768 & -0,550 \\ -0,768 & -0,121 & 0,629 \end{pmatrix} R_1 = \begin{pmatrix} -9,110 & -8,671 & -17,672 \\ 0,000 & -2,794 & -2,777 \\ 0,000 & 0,000 & 1,729 \end{pmatrix}$$

Budoucí matice vlastních čísel a vlastních vektorů vypadají po první iteraci takto:

$$D = \begin{pmatrix} 21,434 & 3,054 & -1,328 \\ 3,054 & 2,480 & -0,209 \\ -1,328 & -0,209 & 1,086 \end{pmatrix} V = \begin{pmatrix} -0,549 & 0,630 & -0,550 \\ -0,329 & -0,768 & -0,550 \\ -0,768 & -0,121 & 0,629 \end{pmatrix}$$

Podívejme se, jak budou tyto matice vypadat po čtyřech iteracích, nejprve však samotný QR rozklad:

$$Q_4 = \begin{pmatrix} -1,000 & 0,001 & 0,000 \\ -0,001 & -1,000 & 0,000 \\ 0,000 & 0,000 & 1,000 \end{pmatrix} R_4 = \begin{pmatrix} -22,000 & -0,028 & 0,003 \\ 0,000 & -2,000 & 0,000 \\ 0,000 & 0,000 & 1,000 \end{pmatrix}$$

A nyní již samotné matice  $D$  a  $V$ :

$$D = \begin{pmatrix} 22,000 & 0,002 & 0,000 \\ 0,002 & 2,000 & 0,000 \\ 0,000 & 0,000 & 1,000 \end{pmatrix} V = \begin{pmatrix} 0,408 & -0,707 & -0,577 \\ 0,408 & -0,707 & -0,577 \\ 0,817 & 0,000 & 0,577 \end{pmatrix}$$

Dostáváme v podstatě stejný výsledek jako v případě Jacobiho metody s tím, že vlastní čísla a vektory jsou v maticích  $D$  a  $V$  seřazeny v jiném pořadí. Dvojice vlastní číslo-vlastní vektor však zůstávají zachovány. ■

Existují různé modifikace QR algoritmu, které urychlují jeho konvergenci. Jednou z nich je úprava matice na třídiagonální tvar před započítáním samotného algoritmu. K takovéto úpravě lze využít právě Householderových transformací aplikovaných zleva i zprava. Dojde tak k vynulování prvků pod subdiagonálou a napravo od superdiagonály. Transformace je nutné zaznamenat do matice  $Q$  následujícím způsobem:

$$Q = P_1 P_2 \dots P_{n-1}. \quad (3.84)$$

Tato matice pak vstupuje do samotného QR algoritmu namísto jednotkové matice  $I$  a na konci algoritmu bude ve svých sloupcích obsahovat vlastní vektory.

QR algoritmus lze zobecnit i pro nesymetrické matice. V takovém případě matice  $A$  postupně nekonverguje k diagonální matici, ale k matici horní trojúhelníkové. To nám však nevadí, jelikož vlastní čísla horní trojúhelníkové matice  $R$  se nacházejí přímo na diagonále. Charakteristický polynom matice  $R$  je totiž roven

$$R - \lambda I = (R_{11} - \lambda)(R_{22} - \lambda) \dots (R_{nn} - \lambda). \quad (3.85)$$

Vlastní vektory lze poté nalézt např. řešením soustavy

$$(R - \lambda_i I)v = o, \quad (3.86)$$

kde  $o$  je nulový vektor a  $\lambda_i$  je jedno z vlastních čísel.

### 3.5 Srovnání metod

Pokusíme se shrnout a porovnat výše popsané metody. Hledání vlastních čísel pomocí charakteristického polynomu není příliš efektivní a jak již bylo uvedeno, v praxi se příliš nepoužívá. Nalezení charakteristického polynomu však může být užitečné, jestliže potřebujeme vypočítat inverzní matici, zvláště pak při posunu spektra. Tím se dostáváme k mocninné metodě. Ta je vhodná hlavně tehdy, když nepotřebujeme nalézt všechna vlastní čísla, ale pouze některá, např. dominantní vlastní číslo, v absolutní hodnotě nejmenší vlastní číslo nebo vlastní číslo nejblíže zadané hodnotě. Rychlost konvergence mocninné metody se může pro různé typy matic výrazně lišit. V tabulce 1 je ilustrována závislost rychlosti konvergence na podílu v absolutní hodnotě druhého největšího vlastního čísla a dominantního vlastního čísla. V tabulce je vždy uveden průměr ze sta měření,

Požadovaná přesnost	Počet iterací			
	$\frac{\lambda_2}{\lambda_1} = 0,969$	$\frac{\lambda_2}{\lambda_1} = 0,969$	$\frac{\lambda_2}{\lambda_1} = 0,082$	$\frac{\lambda_2}{\lambda_1} = 0,082$
0,1	63		3	
0,01	151		4	
0,001	220		5	

Tabulka 1: Závislost počtu iterací mocninné metody na podílu v abs. hodnotě dvou největších vlastních čísel

kdy byla použita jedna matice a v každém měření byl zvolen náhodný počáteční vektor.

Je patrné, že pokud jsou dvě v absolutní hodnotě největší vlastní čísla blízká, konverguje mocninná metoda velmi pomalu, zatímco při velmi malém podílu nalezne dominantní vlastní vektor již po pár iteracích. Výhodou mocninné metody je její použitelnost pro velké řídké matice (při výpočtu nedochází k jejím úpravám), které dostáváme často při řešení různých problémů z mechaniky, např. ve stavebnictví nebo ve strojním průmyslu. Složité struktury se namodelovávají a hledají se jejich vibrační frekvence. Nejdůležitější bývají zpravidla v absolutní hodnotě nejmenší vlastní čísla, jelikož se jedná o frekvence, které se mohou vyskytovat v prostředí obklopující tyto objekty. Kdybychom nevzali tyto vibrace v úvahu, mohlo by dojít k poškození struktur, nebo dokonce k jejich destrukci, jak se stalo např. v roce 1940 u mostu v americké Tacomě.

Zcela odlišného charakteru jsou QR metoda a Jacobiho metoda. Jejich použití pro velké řídké matice již není tak vhodné jako v případě mocninné metody, jelikož by v každé iteraci přibývaly nenulové prvky, takže by mohly nastat problémy s nedostatkem paměti. Jejich velkou výhodou je ale skutečnost, že naleznou všechna vlastní čísla a všechny vlastní vektory najednou, není tedy nutné hledat inverzní matice, či posouvat spektrum. QR metoda a Jacobiho metoda se nezanedbatelně liší v počtu iterací nutných k dosažení určité přesnosti, jak ukazuje tabulka 2: Použity byly náhodné symetrické matice o zadaných

Rozměry matice	Počet iterací	
	QR metoda	Jacobiho metoda
$10 \times 10$	88	120
$20 \times 20$	127	545
$50 \times 50$	157	3867
$70 \times 70$	180	7791
$100 \times 100$	185	16377

Tabulka 2: Závislost počtu iterací na použité metodě

rozměrech. Na každou z nich byly aplikovány obě metody s přesností na 3 desetinná místa. V tabulce je vždy uvedena průměrná hodnota z 20 měření. Z tabulky je okamžitě patrné, že QR metoda konverguje rychleji než Jacobiho metoda. Nevýhodou QR metody je ale nutnost v každé iteraci počítat QR rozklad. Naproti tomu sestavení Givensovy matice je značně rychlejší. Rychlost QR algoritmu tedy závisí na konkrétní implementaci QR rozkladu. Počet iterací je u Jacobiho metody závislý hlavně na dimenzích matice, kdežto u QR metody tento počet u stejně rozměrných matic výrazně kolísá. Další výhodou Jacobiho metody oproti QR metodě je možnost její paralelizace.

## 4 Aplikace metod hledání vlastních čísel

V této kapitole se podíváme na několik oblastí, ve kterých se vyskytují problémy vlastních čísel. V každé podkapitole bude uvedeno seznámení s danou oblastí, nalezení problému vlastních čísel a konkrétní příklad včetně komentáře ohledně možnosti, či nutnosti použití numerických metod probraných v minulé kapitole.

### 4.1 Mechanické kmity

Jednou z oblastí, které směřují ke hledání vlastních čísel je mechanické kmitání. Ukážeme si nejprve jednoduchý příklad jednoho závaží zavěšeného na pružině. Poté provedeme rozšíření na více závaží a pružin.

#### 4.1.1 Jedno závaží na pružině

Představme si závaží o hmotnosti  $m$  zavěšené na lineární pružině o tuhosti  $k$ , viz. obr. 1. Pro takovou soustavu platí pohybová rovnice

$$m\ddot{y} + c\dot{y} + ky = F(t), \quad (4.1)$$

kde  $c$  je koeficient tlumení,  $y$  udává výchylku od rovnovážné polohy a  $F(t)$  je budící síla, jejíž velikost závisí na čase  $t$ . My se budeme zabývat tzv. volnými kmity, kdy nedochází k tlumení, ani k vynuceným kmitům způsobených budící silou. Diferenciální rovnice (4.1) tak přechází v

$$m\ddot{y} + ky = 0. \quad (4.2)$$

Předpokládejme řešení této rovnice ve tvaru

$$y(t) = A \sin(\omega t + \phi_0), \quad (4.3)$$

kde  $\omega$  udává úhlovou frekvenci a  $\phi_0$  počáteční fázi,  $A$  je amplituda kmitů. Potom pro druhou derivaci  $y$  podle času platí

$$\ddot{y}(t) = -\omega^2 A \sin(\omega t + \phi_0). \quad (4.4)$$

Dosazením do rovnice (4.2) získáme rovnici

$$-m\omega^2 A \sin(\omega t + \phi_0) + kA \sin(\omega t + \phi_0) = 0. \quad (4.5)$$

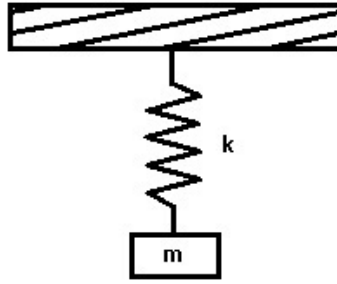
Z této rovnice jednoduše vyjádříme kruhovou frekvenci

$$\omega = \sqrt{\frac{k}{m}}. \quad (4.6)$$

Pro vlastní frekvenci  $f$  platí vztah

$$f = \frac{\omega}{2\pi} = \frac{1}{2\pi} \sqrt{\frac{k}{m}}. \quad (4.7)$$

Nebude-li působit žádná vnější síla, ani tlumení, bude soustava kmitat právě s frekvencí  $f$ . Amplituda kmitání  $A$  bude záviset na počátečním rozkyvu.



Obrázek 1: Závaží na pružině

#### 4.1.2 Více závaží a pružin

Naši soustavu nyní rozšíříme o další závaží a pružiny. Na každém závaží je zavěšena další pružina se závažím. Dohromady máme nyní soustavu  $n$  závaží a  $n$  pružin, poslední závaží je zavěšeno volně, viz. obr. 2. Tuto soustavu můžeme popsat  $n$  diferenciálními rovnicemi o  $n$  neznámých  $y_i$ ,  $i \in \{1, 2, \dots, n\}$ . Při sestavování rovnice pro  $i$ -té závaží již musíme vzít v úvahu výchylky závaží s indexy  $i - 1$  a  $i + 1$ . Soustava rovnic pro  $n$  závaží vypadá následovně:

$$m_1 \ddot{y}_1 + k_1 y_1 + k_2 (y_1 - y_2) = 0 \quad (4.8)$$

$$m_i \ddot{y}_i + k_i (y_i - y_{i-1}) + k_{i+1} (y_i - y_{i+1}) = 0, \quad i = \{2, \dots, n-1\} \quad (4.9)$$

$$m_n \ddot{y}_n + k_n (y_n - y_{n-1}) = 0 \quad (4.10)$$

Roznásobme nyní závorky obsahující  $y_i$  a seskupme členy  $y_i k_j$  tak, aby bylo možno vytknout právě  $y_i$ . Takto upravená soustava vypadá následovně:

$$m_1 \ddot{y}_1 + (k_1 + k_2) y_1 - k_2 y_2 = 0 \quad (4.11)$$

$$m_i \ddot{y}_i - k_i y_{i-1} + (k_i + k_{i+1}) y_i - k_{i+1} y_{i+1} = 0, \quad i = \{2, \dots, n-1\} \quad (4.12)$$

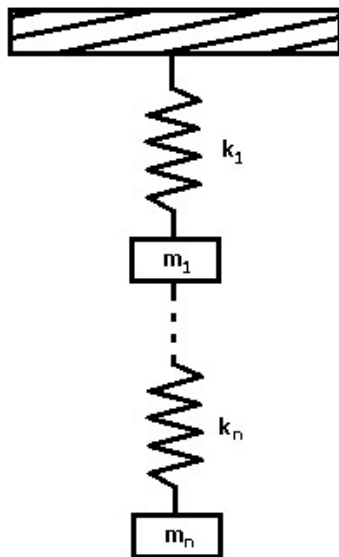
$$m_n \ddot{y}_n - k_n y_{n-1} + k_n y_n = 0 \quad (4.13)$$

Soustavu v tomto tvaru lze zapsat i maticově

$$M \ddot{y} + K y = 0, \quad (4.14)$$

kde

$$M = \begin{pmatrix} m_1 & 0 & \dots & 0 \\ 0 & m_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & m_n \end{pmatrix}, \quad K = \begin{pmatrix} k_1 + k_2 & -k_2 & 0 & \dots & \dots & 0 \\ -k_2 & k_2 + k_3 & -k_3 & & & \vdots \\ 0 & -k_3 & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & 0 \\ \vdots & & & \ddots & k_{n-1} + k_n & -k_n \\ 0 & \dots & \dots & 0 & -k_n & k_n \end{pmatrix}.$$



Obrázek 2: Soustava závaží a pružin

Předpokládáme řešení této soustavy ve tvaru

$$y_i(t) = A_i \sin(\omega t + \phi_0), \quad i \in \{1, 2, \dots, n\}, \quad (4.15)$$

s druhou derivací podle času

$$\ddot{y}_i(t) = -\omega^2 A_i \sin(\omega t + \phi_0), \quad i \in \{1, 2, \dots, n\}. \quad (4.16)$$

Řekněme, že počáteční fáze  $\phi_0$  je rovna  $\frac{\pi}{2}$ , což znamená, že čas  $t$  měříme od okamžiku, kdy uvolníme všechna závaží najednou z počátečních poloh. Potom amplitudy kmitů jsou rovny právě počátečním rozkvyům  $y_{0i}$ . Dosadíme nyní předpokládané řešení do soustavy (4.14):

$$-\omega^2 M y_0 \sin\left(\omega t + \frac{\pi}{2}\right) + K y_0 \sin\left(\omega t + \frac{\pi}{2}\right) = 0. \quad (4.17)$$

Vytknutím funkce  $\sin\left(\omega t + \frac{\pi}{2}\right)$  dostaneme

$$(-\omega^2 M y_0 + K y_0) \sin\left(\omega t + \frac{\pi}{2}\right) = 0. \quad (4.18)$$

Levá strana rovnice musí být nulová nezávisle na čase  $t$ , budeme tudíž řešit rovnici

$$-\omega^2 M y_0 + K y_0 = 0. \quad (4.19)$$

Provedeme nyní tyto úpravy:

$$K y_0 = \omega^2 M y_0 / M^{-1}. \quad (4.20)$$

$$M^{-1} K y_0 = \omega^2 y_0 / \omega^2 = \lambda \quad (4.21)$$

$$M^{-1} K y_0 = \lambda y_0. \quad (4.22)$$

Z poslední rovnice je již patrný problém hledání vlastních čísel. Pokud vypočteme vlastní čísla matice  $M^{-1}K$ , získáme následně vlastní frekvence kmitání soustavy ze vztahu

$$f_i = \frac{\omega_i}{2\pi} = \frac{\sqrt{\lambda_i}}{2\pi}. \quad (4.23)$$

Ke každé vlastní frekvenci připadá vlastní vektor počátečních výchylek  $y_{0i}$ . Tyto vektory se nazývají též modální vektory, jelikož udávají jednotlivé módy (tvary) kmitání, pokud celá soustava kmitá s vlastní frekvencí  $\omega_i$ . Poměry prvků modálního vektoru udávají poměry příslušných amplitud jednotlivých závaží. Obecné kmitý soustavy lze vyjádřit jako lineární kombinaci vlastních kmitů

$$y(t) = \sum_{i=1}^n \alpha_i y_{0i} \sin(\omega_i t + \phi_{0i}), \quad \alpha_i \in \mathbb{R}. \quad (4.24)$$

Za povšimnutí stojí fakt, že matice  $M^{-1}K$  není obecně symetrická. Důsledkem toho nemůžeme s maximální efektivitou využít některé metody hledání vlastních čísel, např. nelze vůbec použít Jacobiho metoda nebo pomocí QR metody bychom našli pouze vlastní čísla, ale vlastní vektory by se daly najít pouze řešením soustavy, ne přímo QR metodou. Tento problém se dá jednoduše obejít úpravou rovnice (4.20):

$$Ky_0 = \omega^2 My_0 \quad (4.25)$$

$$Ky_0 = \omega^2 M^{\frac{1}{2}} M^{\frac{1}{2}} y_0 / M^{-\frac{1}{2}}. \quad (4.26)$$

$$M^{-\frac{1}{2}} Ky_0 = \omega^2 M^{\frac{1}{2}} y_0 \quad (4.27)$$

$$M^{-\frac{1}{2}} K M^{-\frac{1}{2}} M^{\frac{1}{2}} y_0 = \omega^2 M^{\frac{1}{2}} y_0 \quad (4.28)$$

$$M^{-\frac{1}{2}} K M^{-\frac{1}{2}} M^{\frac{1}{2}} y_0 - \omega^2 M^{\frac{1}{2}} y_0 = 0 \quad (4.29)$$

$$(M^{-\frac{1}{2}} K M^{-\frac{1}{2}} - \omega^2 I) M^{\frac{1}{2}} y_0 = 0 / M^{\frac{1}{2}} y_0 = v \quad (4.30)$$

$$(M^{-\frac{1}{2}} K M^{-\frac{1}{2}} - \omega^2 I) v = 0. \quad (4.31)$$

Vzhledem k tomu, že matice  $M$  je diagonální, matice  $M^{-\frac{1}{2}} K M^{-\frac{1}{2}}$  je symetrická. Z poslední rovnice je patrné, že matice  $M^{-\frac{1}{2}} K M^{-\frac{1}{2}}$  má stejná vlastní čísla, jako matice  $M^{-1}K$ . Vlastní vektory matice  $M^{-\frac{1}{2}} K M^{-\frac{1}{2}}$  jsou rovny  $v = M^{\frac{1}{2}} y_0$ . Chceme-li najít vlastní vektory matice  $M^{-1}K$ , stačí vektory  $v$  přenásobit zleva  $M^{-\frac{1}{2}}$ :

$$y_0 = M^{-\frac{1}{2}} v. \quad (4.32)$$

Ukážeme si nyní konkrétní příklad.

#### Příklad 4.1

Nalezneme vlastní frekvence a jím příslušné módy kmitání soustavy pružin, kde

$$m_1 = m_2 = m_3 = m_4 = 4 \text{ kg}$$

$$m_5 = 9 \text{ kg}$$

$$k_1 = k_2 = k_3 = k_4 = k_5 = 2 \text{ N} \cdot \text{m}^{-1}.$$

Sestavíme příslušné matice hmotnosti a tuhosti:

$$M = \begin{pmatrix} 4 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 9 \end{pmatrix}, \quad K = \begin{pmatrix} 4 & -2 & 0 & 0 & 0 \\ -2 & 4 & -2 & 0 & 0 \\ 0 & -2 & 4 & -2 & 0 \\ 0 & 0 & -2 & 4 & -2 \\ 0 & 0 & 0 & -2 & 2 \end{pmatrix}.$$

Jelikož matice  $M^{-1}K$  je nesymetrická, použijeme k nalezení vlastních čísel matici

$$M^{-\frac{1}{2}}KM^{-\frac{1}{2}}.$$

Potom vlastní frekvence určíme jako  $\omega_i = \sqrt{\lambda_i}$  a módy kmitání jako  $y_{0i} = M^{\frac{1}{2}}v_i$ . K výpočtu použijeme Jacobiho metodu s přesností na 3 desetinná místa. Výsledek po 22 iteracích vypadá následovně:

$$\begin{aligned} \omega_1 &= 1,349 \text{ s}^{-1}, \quad \omega_2 = 1,160 \text{ s}^{-1}, \quad \omega_3 = 0,871 \text{ s}^{-1}, \quad \omega_4 = 0,520 \text{ s}^{-1}, \quad \omega_5 = 0,166 \text{ s}^{-1}, \\ y_{01} &= \begin{pmatrix} -0,355 \\ 0,582 \\ -0,599 \\ 0,399 \\ -0,126 \end{pmatrix}, \quad y_{02} = \begin{pmatrix} 0,570 \\ -0,395 \\ -0,297 \\ 0,600 \\ -0,267 \end{pmatrix}, \quad y_{03} = \begin{pmatrix} -0,560 \\ -0,271 \\ 0,429 \\ 0,478 \\ -0,446 \end{pmatrix}, \quad y_{04} = \begin{pmatrix} 0,350 \\ 0,511 \\ 0,395 \\ 0,065 \\ -0,675 \end{pmatrix}, \\ y_{05} &= \begin{pmatrix} 0,098 \\ 0,191 \\ 0,273 \\ 0,340 \\ 0,874 \end{pmatrix}. \end{aligned}$$

Vypočítali jsme tedy vlastní frekvence a módy kmitání zadané soustavy. ■

## 4.2 Molekulová dynamika

Asi nejjednodušším příkladem z oblasti molekulové dynamiky jsou vibrace částicových soustav v harmonické aproximaci (viz. [9], [10]). Podívejme se nejprve na jednu částici o hmotnosti  $m$ . Schrödingerova rovnice pro jednočásticový harmonický oscilátor má následující tvar:

$$\left[ -\frac{\hbar^2}{2m} \frac{d^2}{dr^2} + V(r) \right] \psi(r) = E\psi(r). \quad (4.33)$$

Interakční potenciál  $u(r)$  aproximujeme jeho Taylorovým rozvojem druhého řádu kolem jeho minima

$$u(r) \approx u(r_0) + \frac{du}{dr}(r_0)x + \frac{1}{2} \frac{d^2u}{dr^2}(r_0)x^2, \quad (4.34)$$



kde  $x = r - r_0$ . Druhý člen Taylorova rozvoje je roven 0 (v minimu potenciálu je jeho první derivace nulová). Zároveň platí rovnost  $u(x_0) = V(x_0)$  (minimální celková potenciální energie je rovna minimálnímu interakčnímu potenciálu). Po dosazení  $u(r)$  do rovnice (4.33) za potenciální energii  $V(x)$  dostáváme

$$\left[ -\frac{\hbar^2}{2m} \frac{d^2}{dx^2} + \frac{1}{2} \frac{d^2 u}{dr^2}(r_0) x^2 \right] \psi(r) = [E - V(r_0)] \psi(r). \quad (4.35)$$

Pro vlastní energetické hladiny  $E_n$ , které náleží řešením této rovnice, platí

$$E_n - V(r_0) = \hbar \omega \left( n + \frac{1}{2} \right), \quad n = 0, 1, 2, \dots, \quad \omega = \sqrt{\frac{\frac{d^2 u}{dr^2}(r_0)}{m}}. \quad (4.36)$$

Úhlová frekvence kmitů částice je rovna  $\omega$ . Číslo  $n$  udává kvantové číslo energetické hladiny.

Problém lze analogicky rozšířit pro třírozměrné vícečásticové soustavy. Pro vlastní energetické hladiny takovéto soustavy platí:

$$E_{n_{11}n_{12}n_{13}n_{21}\dots} - V(r_0) = \hbar \sum_{i=1}^N \sum_{j=1}^3 \omega_{ij} \left( n_{ij} + \frac{1}{2} \right), \quad n_{ij} = 0, 1, 2, \dots, \quad \omega_{ij} = \sqrt{\frac{k_{ij}}{M_i}}, \quad (4.37)$$

kde  $k_{ij}$  jsou vlastní čísla Hessovy matice potenciálu (matice obsahující druhé derivace potenciálu vyčíslené v minimu potenciálu),  $N$  udává počet atomů,  $M_i$  je tzv. redukovaná hmotnost. Redukovanou hmotnost použijeme tehdy, pokud pohyb soustavy několika částic řešíme jako pohyb jedné částice. Jestliže jsou částice stejné, pak je rovna hmotnosti jedné částice dělené počtem částic. Pokud soustavu nepřevádíme na jednu částici a částice jsou stejné, dosadíme za  $M_i$  hmotnost jedné částice.  $r$  je vektor souřadnic

$$r = \begin{pmatrix} x_1 \\ y_1 \\ z_1 \\ x_2 \\ \vdots \\ z_N \end{pmatrix}. \quad (4.38)$$

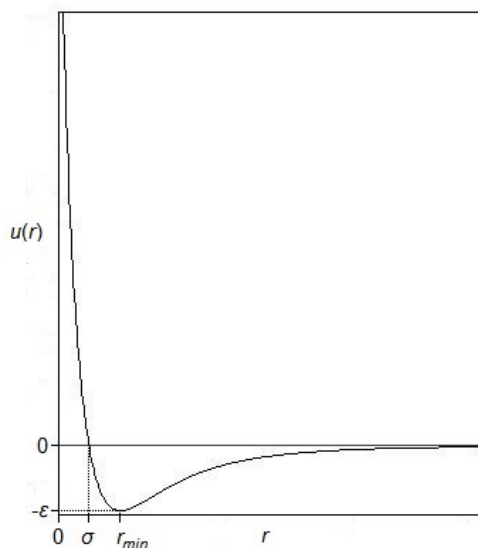
V případě, že všechna kvantová čísla  $n_{ij}$  jsou nulová, nachází se soustava ve stavu s nejnižší energií, tzv. energií základního stavu. Rozdíl  $E_{000\dots} - V(r_0)$  se nazývá energií nulových kmitů a jedná se o nejnižší energii, které může soustava nabývat.

Ukážeme si příklad na jednoduché dvouatomové struktuře.

#### Příklad 4.2

Vypočteme vibrační frekvence soustavy dvou atomů argonu. Pro dvouatomové molekuly lze jako potenciál použít tzv. Lennard-Jonesův potenciál. Jeho vzorec vypadá následovně:

$$u(r) = 4\epsilon \left[ \left( \frac{\sigma}{r} \right)^{12} - \left( \frac{\sigma}{r} \right)^6 \right], \quad (4.39)$$



Obrázek 3: Graf Lennard-Jonesova potenciálu

kde parametr  $\epsilon$  udává hloubku minima (minimální hodnotu potenciálu) a parametr  $\sigma$  vzdálenost, při které je potenciál nulový. Potenciál nabývá minima ve vzdálenosti  $r = 2^{\frac{1}{6}}\sigma$ . Graf Lennard-Jonesova potenciálu je na obr. 3. Pro argon jsou tyto hodnoty tabelovány:

$$\begin{aligned}\epsilon &= 83,265 \text{ cm}^{-1} = 83,265 \cdot 1,98658 \cdot 10^{-23} \text{ J}, \\ \sigma &= 3,405 \text{ Å (Angstrom)} = 3,405 \cdot 10^{-10} \text{ m}.\end{aligned}$$

Soustavu atomů umístíme do kartézské souřadnicové soustavy. Potom Lennard-Jonesův potenciál má předpis

$$u(r) = 4\epsilon \left( \left( \frac{\sigma}{r} \right)^{12} - \left( \frac{\sigma}{r} \right)^6 \right), \quad (4.40)$$

$$r = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}. \quad (4.41)$$

Nyní sestavíme Hessovu matici potenciálu a umístíme soustavu částic tak, že potenciál je minimální, v našem případě

$$\begin{pmatrix} x_1 \\ y_1 \\ z_1 \\ x_2 \\ y_2 \\ z_2 \end{pmatrix} = \begin{pmatrix} -\frac{1}{2}2^{\frac{1}{6}}\sigma \\ 0 \\ 0 \\ \frac{1}{2}2^{\frac{1}{6}}\sigma \\ 0 \\ 0 \end{pmatrix}.$$

Vyčíslená Hessova matice vypadá takto:

$$H = \frac{72\epsilon}{\sqrt[3]{2}\sigma^2} \begin{pmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Za zmínku stojí skutečnost, že bychom mohli dva atomy argonu umístit jakkoliv do prostoru. Hessova matice by ale měla více nenulových hodnot a byli bychom k nalezení vlastních čísel nuceni použít nějakou numerickou metodu. Obecně složitější struktury nelze takto výhodně umístit do soustavy souřadnic a nutnosti využít numerických metod bychom se nevyhnuli. Nalézt vlastní čísla naší Hessovy matice lze analyticky. Můžeme však použít i metody numerické. Kdybychom použili např. Jacobiho metodu, skončila by hned po první iteraci. Naše matice má pouze jedno nenulové vlastní číslo rovno

$$\lambda = \frac{144\epsilon}{\sqrt[3]{2}\sigma^2}.$$

Vibrační frekvence struktury je rovna

$$f = \frac{1}{2\pi}\omega = \frac{1}{2\pi}\sqrt{\frac{144\epsilon}{\sqrt[3]{2}\sigma^2 m_{Ar}}} \doteq 7,89 \cdot 10^{11} \text{ s}^{-1}.$$

Vypočítali jsme tedy vibrační frekvenci dimeru argonu. ■

## 5 Závěr

Cílem práce bylo seznámení s rozsáhlou problematikou vlastních čísel a vlastních vektorů. Po stručném úvodu do spektrální teorie byly probrány základní metody hledání vlastních čísel. Vedle jednoduchých příkladů byla diskutována vhodnost jejich použití, a také byly jednotlivé metody srovnány navzájem mezi sebou. Na závěr byly uvedeny oblasti, kde se problémy vlastních čísel vyskytují, včetně jednoduché úlohy z oblasti molekulové dynamiky. Ve skutečnosti je daná problematika mnohem složitější. Dnes již existují efektivnější metody hledání vlastních čísel než zde uvedené. Důvodem pro neustálé zdokonalování těchto metod je postupný nárůst složitosti řešených problémů spojený s rostoucími rozměry vznikajících matic. Pokračování této práce by mělo proniknout hlouběji jak do problematiky molekulové dynamiky, tak do složitějších, ale výkonnějších metod hledání vlastních čísel a vlastních vektorů.

Martin Mrovec

## 6 Reference

- [1] SAAD, Y.: *Numerical methods for large eigenvalue problems*. Rev. ed. Philadelphia: Society for Industrial and Applied Mathematics, c2011, xvi, 276 p. ISBN 978-161-1970-722.
- [2] SCHATZMAN, M.: *A simple proof of convergence of the QR algorithm for normal matrices without shifts*. IMA Preprint Series # 720, 1990.  
Dostupné z: <http://www.ima.umn.edu/preprints/Nov90Series/720.pdf>
- [3] *Numerical recipes: the art of scientific computing*. 3rd ed. Cambridge: Cambridge University Press, 2007, xxi, 1235 s. ISBN 978-0-521-88068-8.
- [4] *The Cayley-Hamilton theorem*. [online]. [2010] [cit. 2013-03-23].  
Dostupné z: <http://math.stanford.edu/~ralph/math53h/notes2.pdf>
- [5] DOSTÁL, Z.: *Lineární algebra*. VŠB-TU Ostrava, Fakulta elektrotechniky a informatiky, 2000
- [6] KOZUBEK, T., BRZOBOHATÝ, T., HAPLA, V., JAROŠOVÁ, M., MARKOPOULOS, A.: *Lineární algebra s Matlabem*. VŠB-TU Ostrava, Fakulta elektrotechniky a informatiky, 2011
- [7] BOUCHALA, J., SADOWSKÁ, M.: *Mathematical Analysis I*. VŠB-TU Ostrava, Fakulta elektrotechniky a informatiky, 2007
- [8] VONDRÁK, V., POSPÍŠIL, L.: *Numerické metody I*. VŠB-TU Ostrava, Fakulta elektrotechniky a informatiky, 2011
- [9] *Vibrace vícečástečkových soustav v harmonické aproximaci*. [online]. [cit. 2013-03-23].  
Dostupné z: <http://artemis.osu.cz/ppmfc/u05.pdf>
- [10] *Vibrace molekul*. [online]. [cit. 2013-03-23].  
Dostupné z: [http://artemis.osu.cz/mm fyz/am/am\\_6\\_5\\_1.htm](http://artemis.osu.cz/mm fyz/am/am_6_5_1.htm)

## A Zdrojové kódy

Níže jsou uvedeny zdrojové kódy algoritmů výpočtu vlastních čísel. Nacházejí se rovněž na příloženém CD.

### A.1 Jacobiho metoda

Zde je uveden zdrojový kód Jacobiho metody implementované v Matlabu.

---

```

function [ eigval,eigvec, iter ] = JacobiDiag( A, myeps )
%eigval—diag.matrix of egenvalues, eigvec—matrix of eigenvectors
%iter — nr of iterations
format long;
iter = 0;
S = A;
V = eye(size(S,1));
[ i , j ] = FindPivotIndex(S);
while(norm(S—diag(diag(S)))>=myeps)
    G = CreateGivensMatrix(S,i,j);
    S = G'*S*G;
    V = V*G;%rotate eigenvectors
    [ i , j ] = FindPivotIndex(S);
    iter = iter + 1;
end
eigvec = V;
eigval = diag(diag(S));

end

function [ p, q ] = FindPivotIndex( A )
format long;
n = size(A,1);
p = 1;
q = 2;
pivot = 0;
for i = 1:n-1
    for j = i+1:n
        if (abs(A(i,j)) > abs(pivot))
            pivot = A(i,j);
            p = i;
            q = j;
        end
    end
end

end

function [ G ] = CreateGivensMatrix( M, i, j )
format long;
if (M(i,i) == M(j,j))
    mytheta = pi/4;
else
```

```
    mytheta = 1/2 * atan( 2 * M(i,j) / (M(i,i) - M(j,j)));  
end  
G = eye(size(M,1));  
G(i,i) = cos(mytheta);  
G(j,j) = cos(mytheta);  
G(i,j) = -sin(mytheta);  
G(j,i) = sin(mytheta);  
  
end
```

---

Výpis 1: Implementace Jacobiho metody v Matlabu

## A.2 QR metoda

Zde je uveden zdrojový kód QR metody implementované v Matlabu.

---

```

function [ D, Q, iter ] = QRMethod( A, myeps, qrsearch )
%D--diag.matrix of egenvalues, Q--matrix of eigenvectors
%iter -- nr of iterations
%qrsearch -- if 0, then uses matlab QR, else use Housholder transformation
D = A;
Q = eye(size(D,1));
iter = 0;
while ((norm(D-diag(diag(D))))>=myeps)&&iter < 2500)
    if qrsearch == 0
        [Qk,Rk] = qr(D);
    else
        [Qk,Rk] = FindQR(D);
    end

    D = Rk*Qk;
    Q = Q*Qk;
    iter = iter + 1;
end
end

function [ Q,R ] = FindQR(A)
%find Q, R using Householder transf.
n = size(A,1);
Q=eye(n);
R=A;
for i=1:n-1
    x=R(i:n,i);
    v=-sign(x(1))*norm(x)*eye(n-i+1,1)-x;
    if (norm(v)>0)
        v=v/norm(v);
        P=eye(n);
        P(i:n,i:n)=P(i:n,i:n)-2*v*v';
        R=P*R;
        Q=Q*P;
    end
end
end
end

```

---

Výpis 2: Implementace QR metody v Matlabu



### A.3 Mocninná metoda

Zde je uveden zdrojový kód mocninné metody implementované v Matlabu.

---

```
function [ lambda, v, it ] = PowerMethod( A, myeps )  
% Searches dominant eigenvector & eigenvalue  
v1 = zeros(size(A,1),1);  
v2 = rand(size(A,1),1);  
lambda = 0;  
it = 0;  
while((norm(A*v2-lambda*v2)>myeps) && it <= 10/myeps)  
    v1 = v2;  
    v2 = A*v1;  
    lambda = v2(1)/v1(1);  
    v2 = v2/norm(v2);  
    it = it+1;  
end  
v = v2;  
if it >= floor(10/myeps)  
    disp( 'Eigenvalue_and_eigenvector_not_found!!');  
end  
end
```

---

Výpis 3: Implementace mocninné metody v Matlabu

## A.4 Mocninná metoda s posunem spektra

Zde je uveden zdrojový kód mocninné metody s posunem spektra implementované v Matlabu.

---

```
function [ lambda, lambdashifted, v, it ] = ShiftedPowerMethod( A, shift, myeps )
% Searches dominant eigenvector & eigenvalue of matrix with shifted
% spectre. lambdashifted is the eigenvalue of shifted matrix, lambda is the
% eigenvalue of matrix A
v1 = zeros(size(A,1),1);
v2 = rand(size(A,1),1);
lambda = 0;
it = 0;
A = A + shift*eye(size(A,1));
while((norm(A*v2-lambda*v2)>myeps) && it <= 10/myeps)
    v1 = v2;
    v2 = A*v1;
    lambda = v2(1)/v1(1);
    v2 = v2/norm(v2);
    it = it+1;
end
v = v2;
lambdashifted = lambda;
lambda = lambda - shift;
if it >= floor(10/myeps)
    disp( 'Eigenvalue_and_eigenvector_not_found!!');
end
end
```

---

Výpis 4: Implementace mocninné metody s posunem spektra v Matlabu

## A.5 Inverzní mocninná metoda

Zde je uveden zdrojový kód modifikace mocninné metody pro nalezení vlastního čísla nejbližší zadané hodnotě implementované v Matlabu.

---

```
function [ lambda, v, it ] = InversePowerMethod( A, value, myeps )
% Searches eigenvector and eigenvalue, that is the closest to value.
v1 = zeros(size(A,1),1);
v2 = rand(size(A,1),1);
lambda = 0;
it = 0;
A = A - value*eye(size(A,1));
if (rank(A) < size(A,1))%if value is eigenvalue, then shift A a bit (myeps)
    A = A+myeps*eye(size(A,1));
end
while((norm(A\v2-lambda*v2)>myeps) && it <= 10/myeps)
    v1 = v2;
    v2 = A\v1;
    lambda = v2(1)/v1(1);
    v2 = v2/norm(v2);
    it = it+1;
end
v = v2;
lambda = 1/lambda + value;
if it >= floor(10/myeps)
    disp( 'Eigenvalue_and_eigenvector_not_found!!');
end
end
```

---

Výpis 5: Implementace nalezení vlastního čísla nejbližšího zadané hodnotě